

The Jini Technology

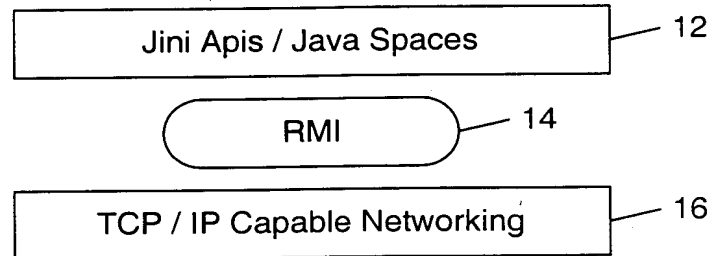


Fig. 1
(Prior Art)

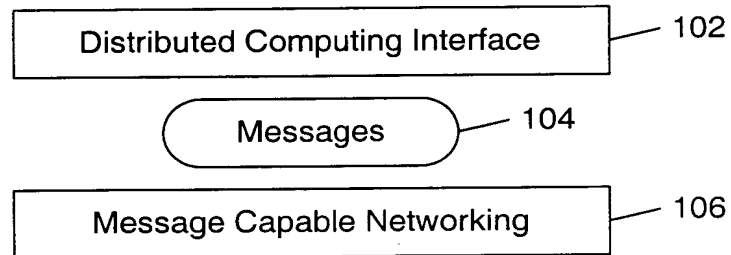


Fig. 2

006701-1226990

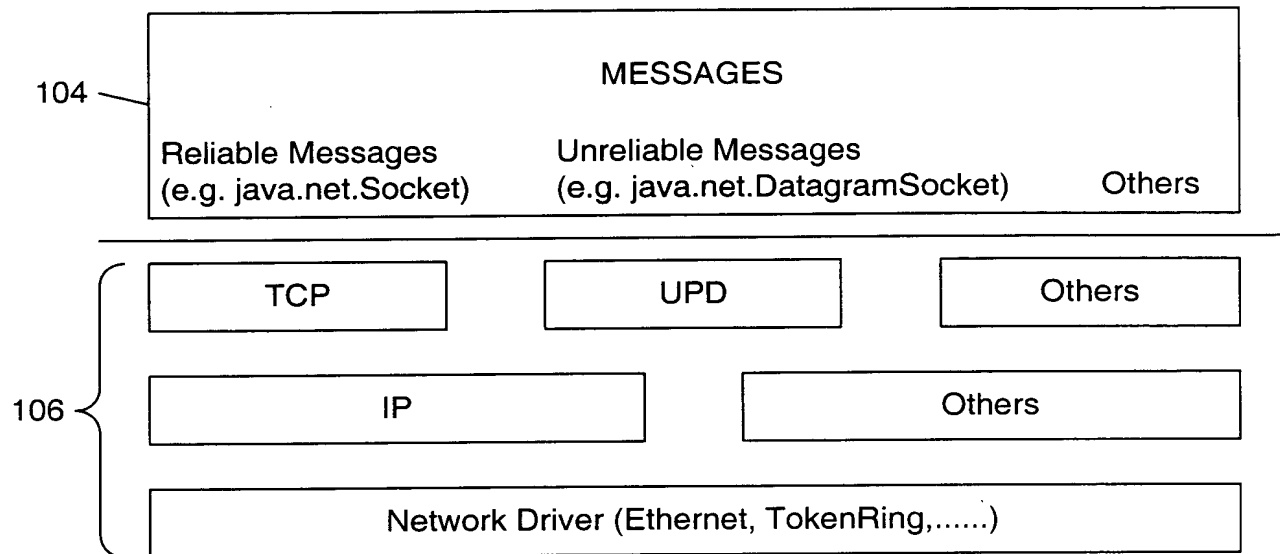


Fig. 3

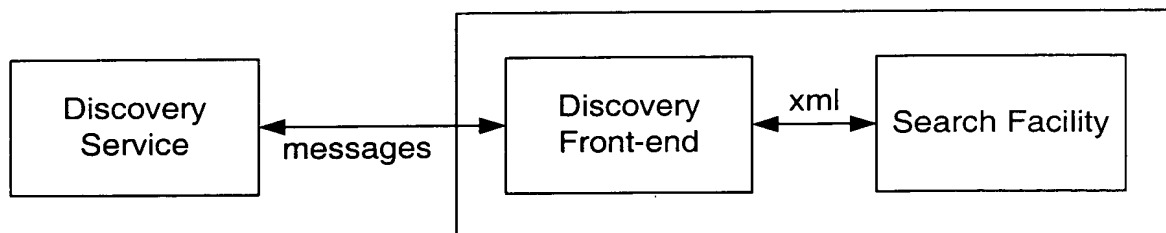


Fig. 4

Dynamic

Client Profile

Claim and Release (a lease) Find (an object)

Write and Take (an object) Notify on Write Public Apis

Dynamic XML to / from Objects

Send / Receive Byte Messages

Connection Transport

Static

Client Profile

Static XML Message Definitions

Send / Receive Byte Messages

Connection Transport

Fig. 5

100-443887-100

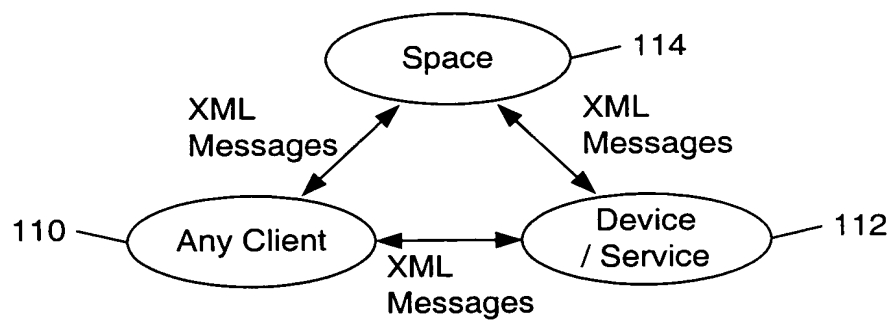


Fig. 6

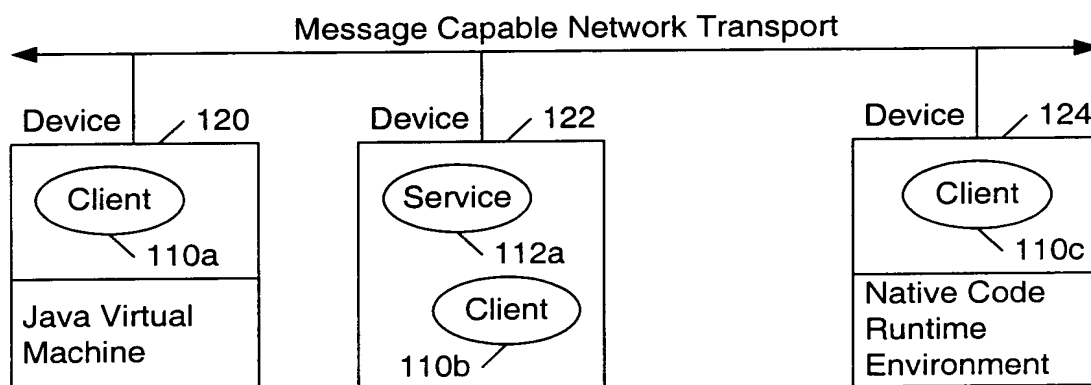


Fig. 7

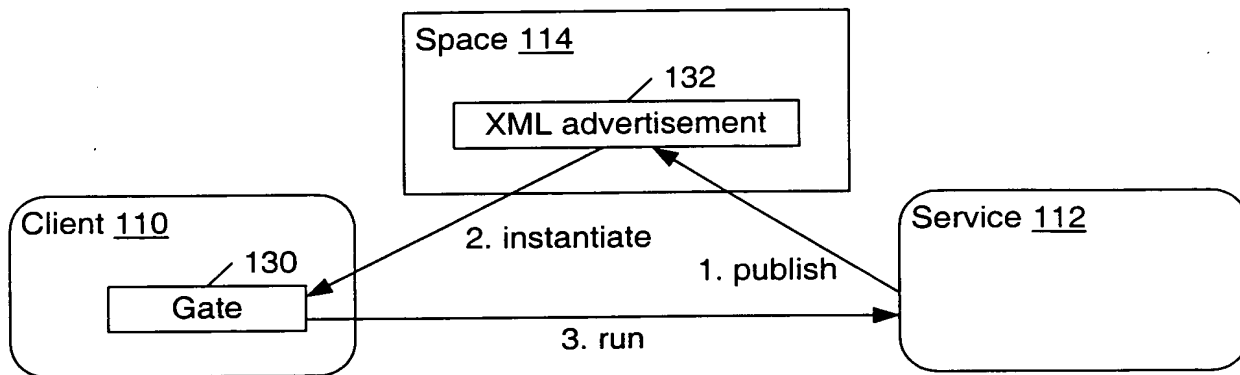


Fig. 8

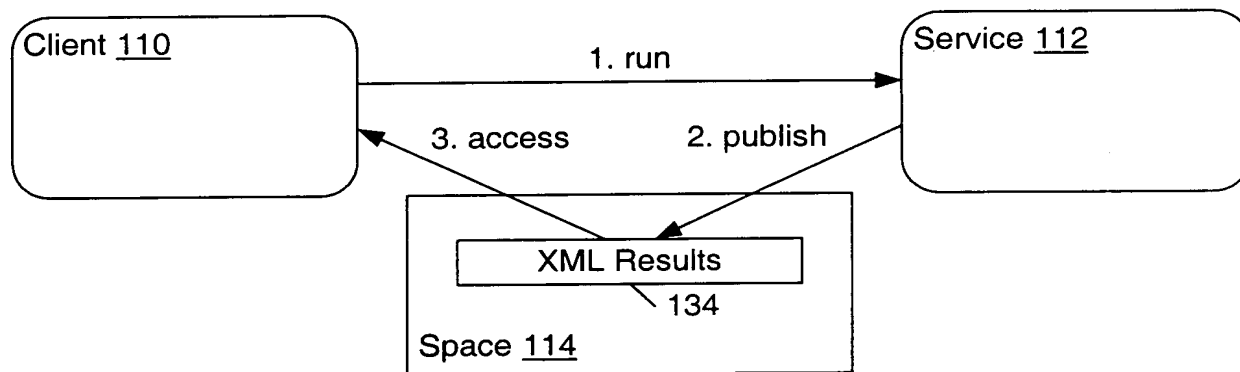


Fig. 9

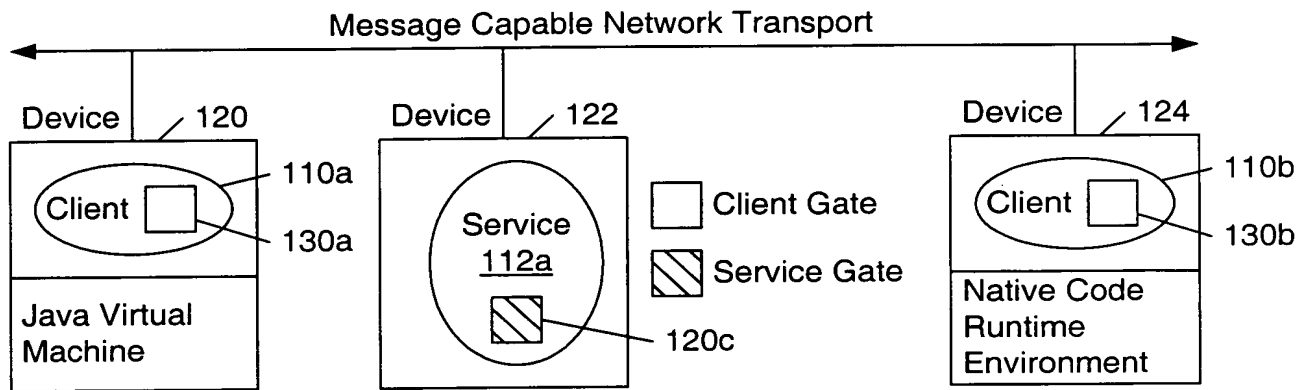


Fig. 10a

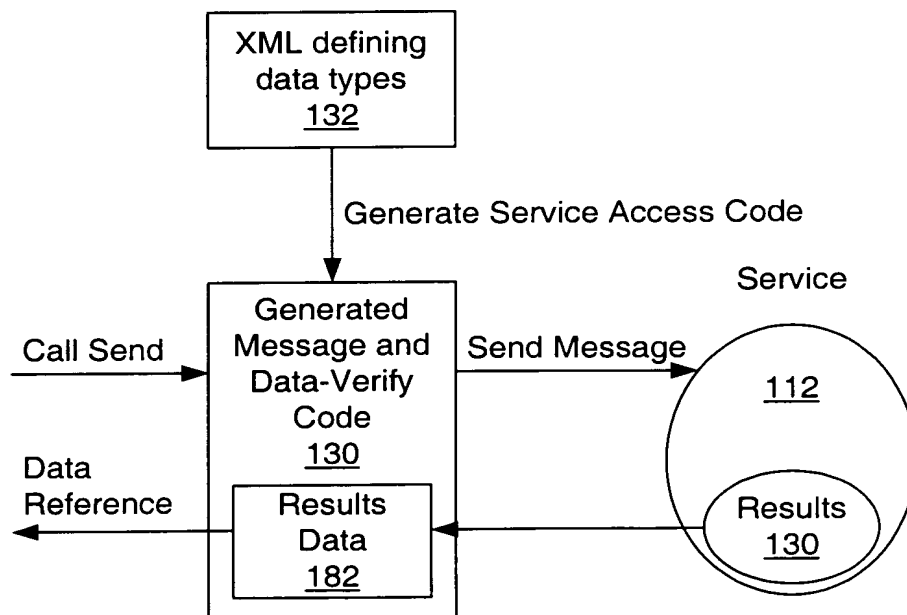


Fig. 10b

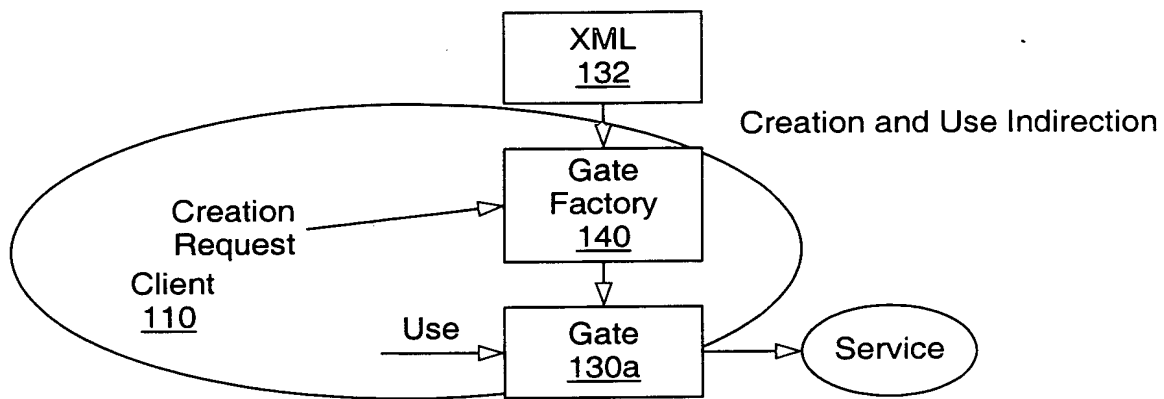


Fig. 11a

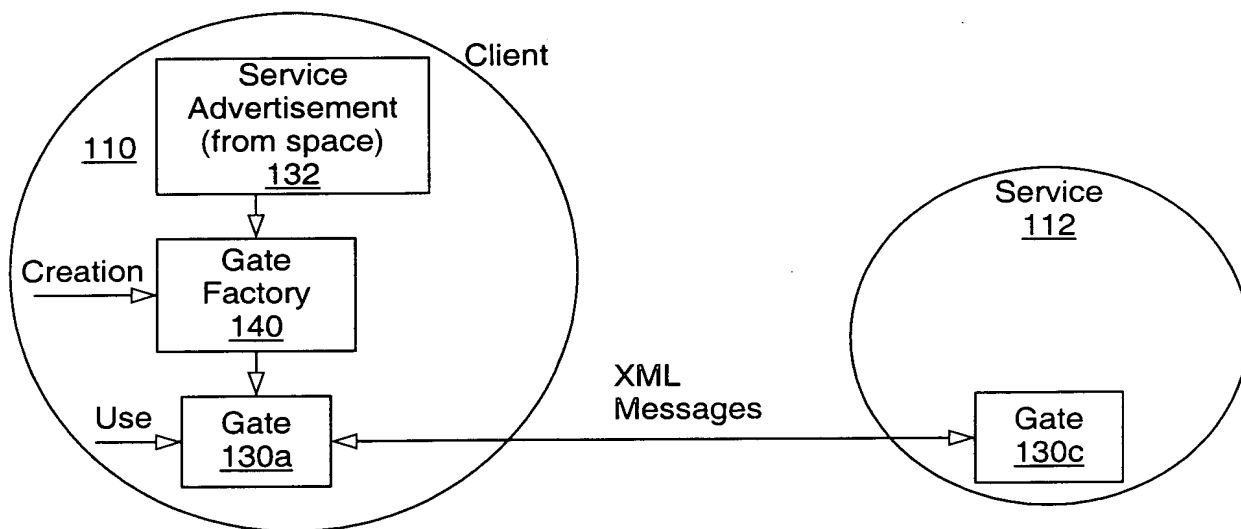


Fig. 11b

The diagram illustrates the Message Gate and Message System architecture. On the left, a large rectangle labeled 'Message Gate' (156) contains several components: 'Key' (150), 'Name' (154), 'Schema' (160), 'Lease' (158), 'Dest. GateName' (152), 'Conductor' (160), and 'Transport Ref.' (153). Below the Message Gate is a 'Message System' block, which is connected to a 'Transport' block. A 'Client or Service' oval is connected to the Message Gate. On the right, a similar stack of 'Message Gate', 'Message System', and 'Transport' blocks is shown, with a double-headed arrow connecting the 'Transport' blocks of both sides.

Fig. 12

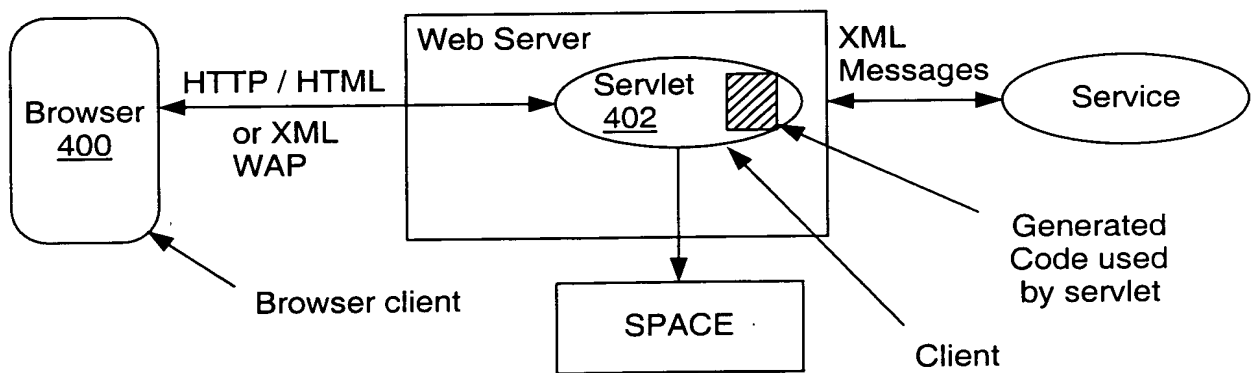


Fig. 13

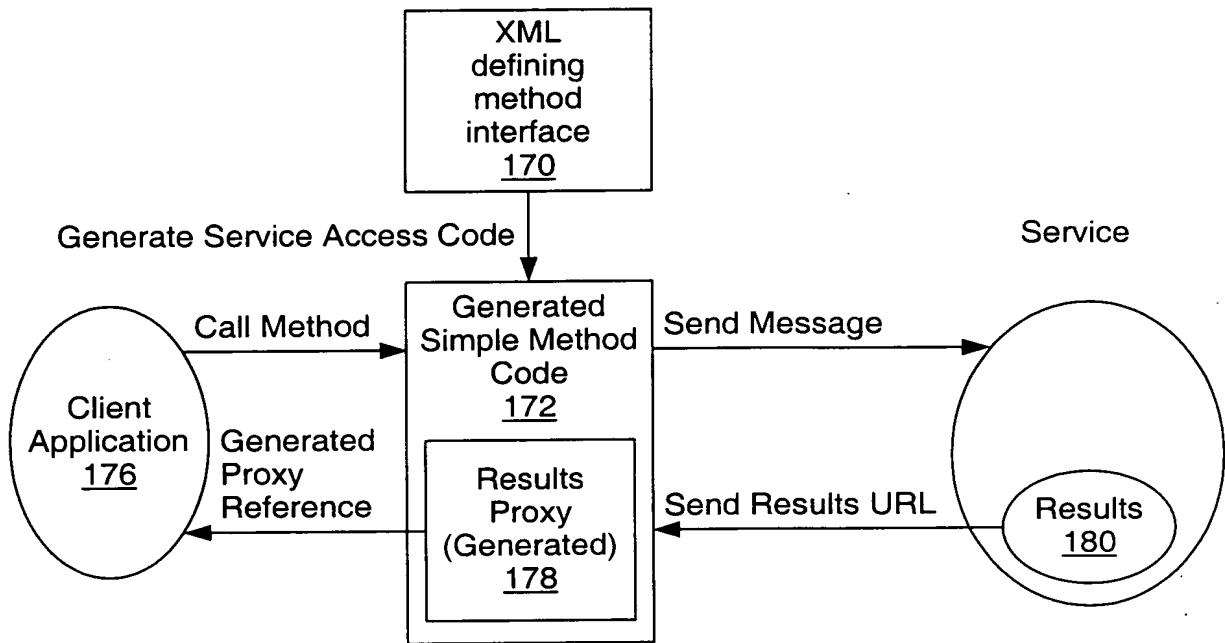


Fig. 14

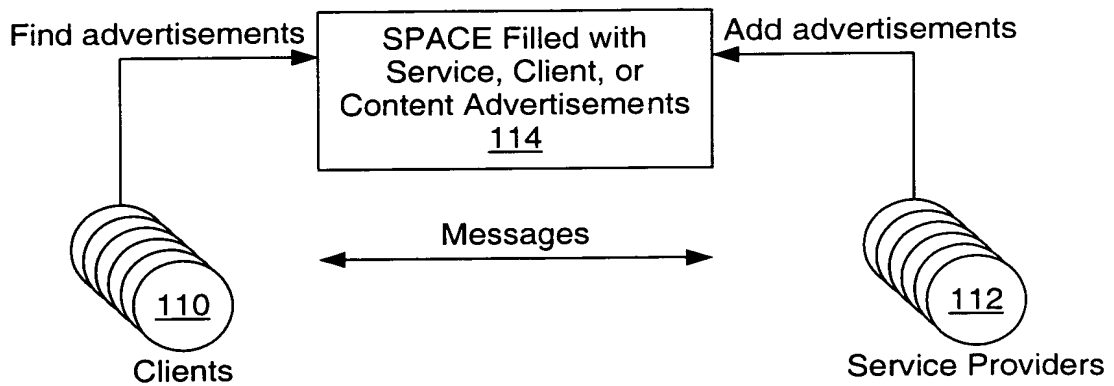


Fig. 15

```
graph LR; Input(( )) -- 1 --> Drafted((Drafted)); Drafted -- 2 --> Published((Published)); Published -- 3 --> Persistent((Persistent)); Persistent -- 4 --> Published; Published -- 5 --> Drafted;
```

Fig. 17

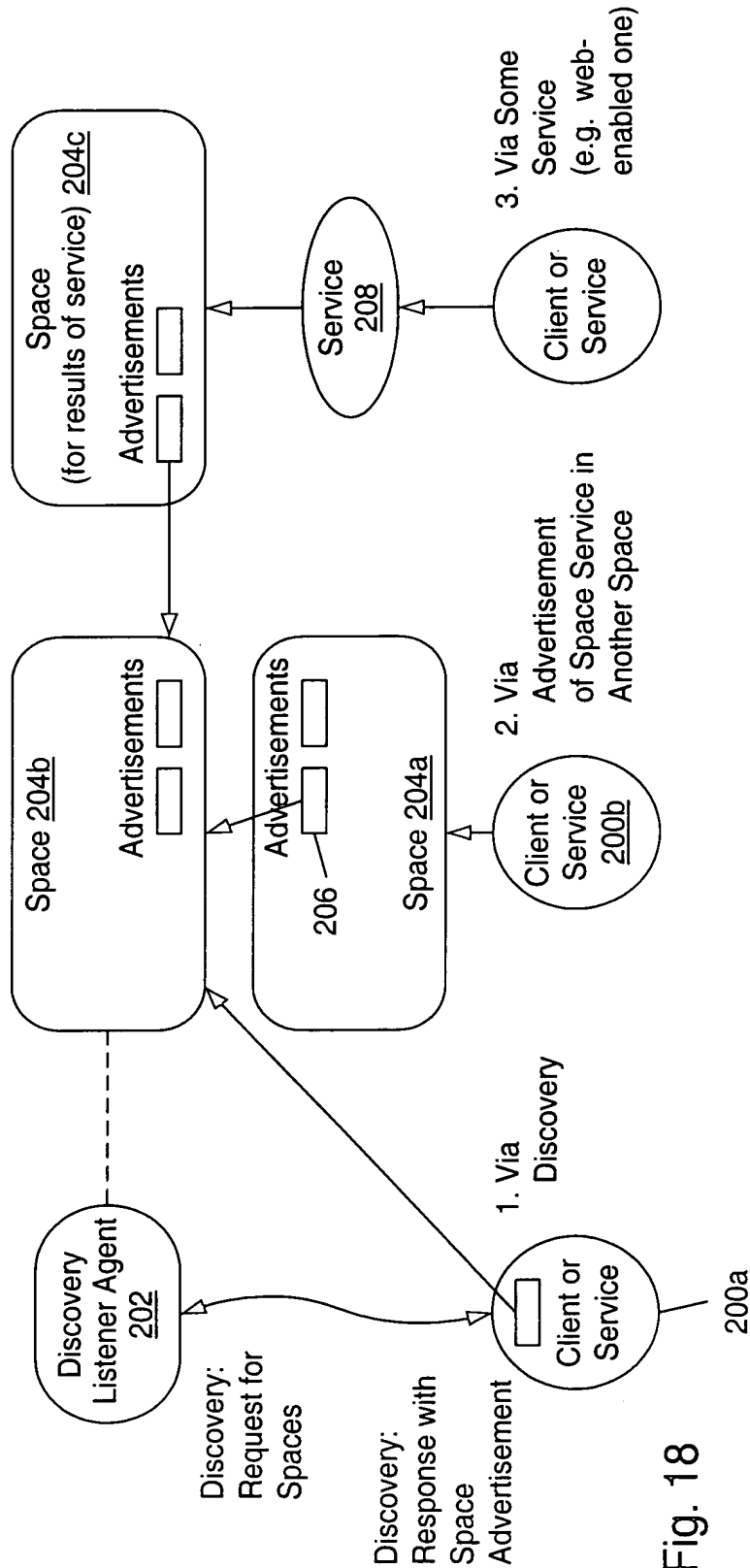


Fig. 18

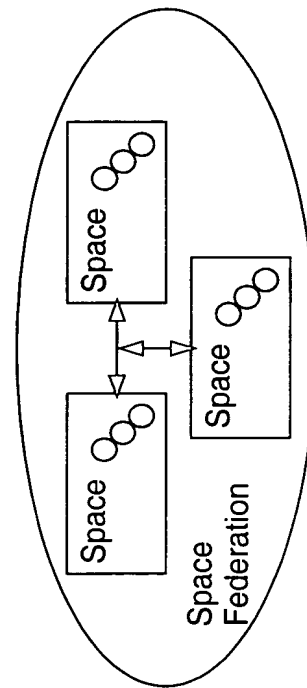


Fig. 19

SECRET

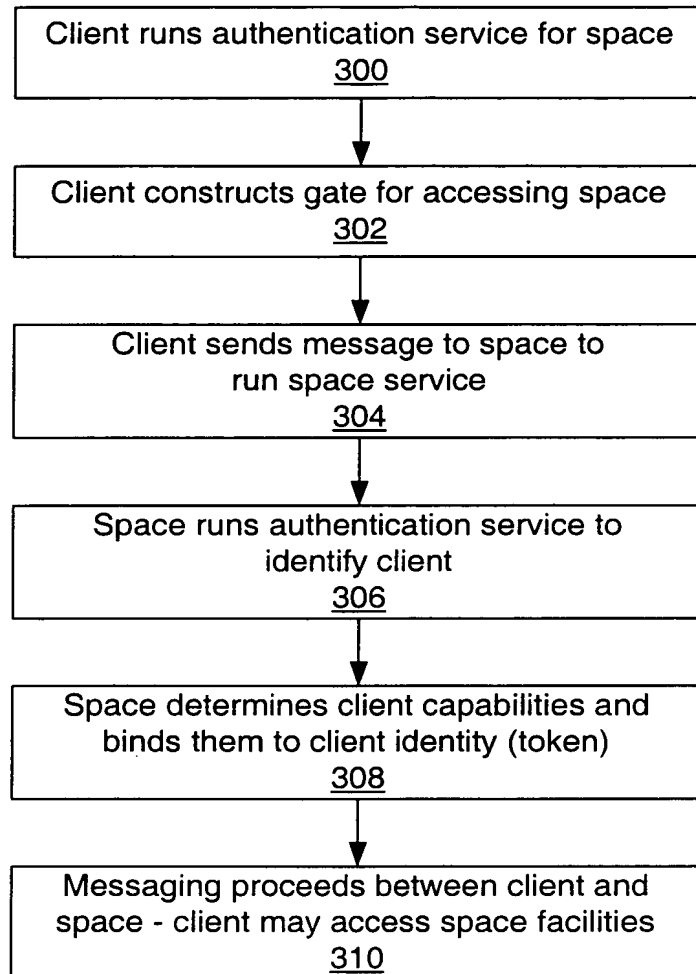


Fig. 20

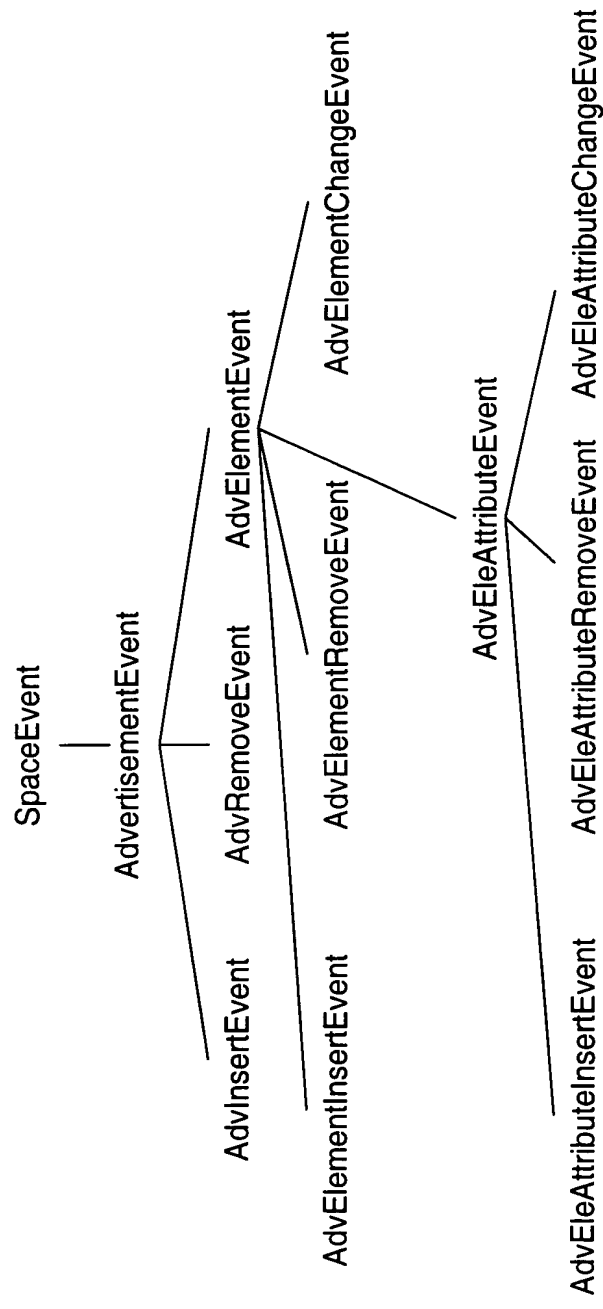


Fig. 21

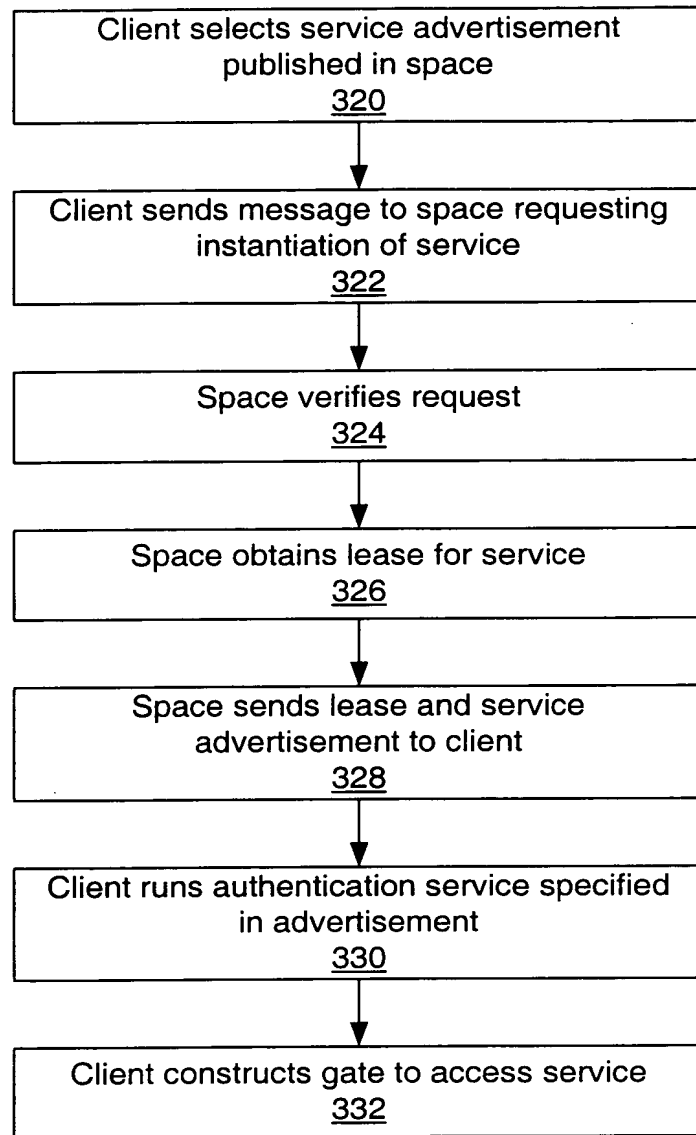


Fig. 22

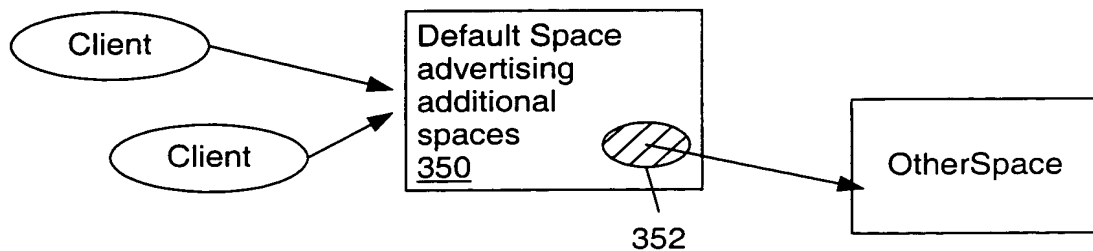


Fig. 23

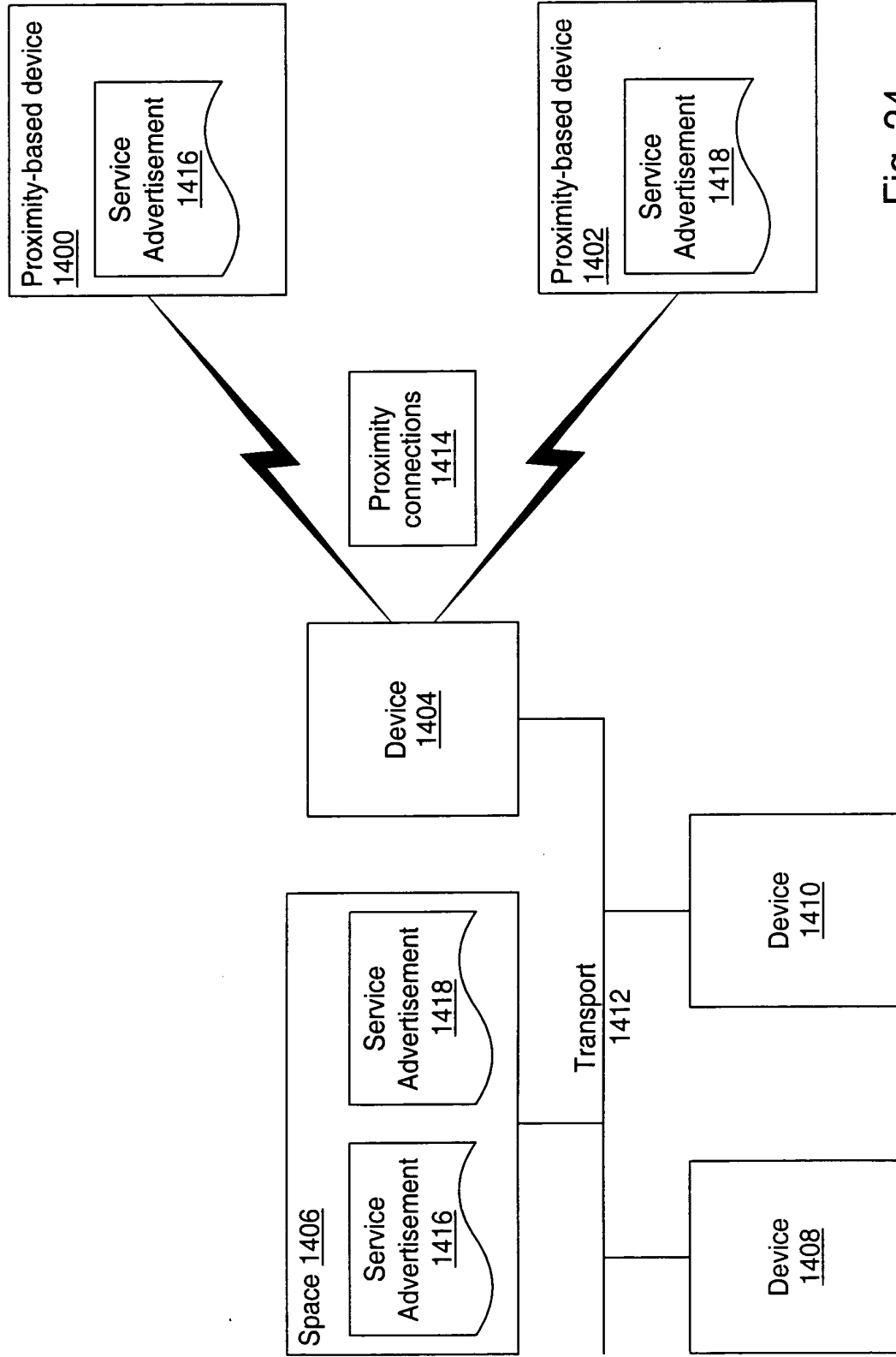


Fig. 24

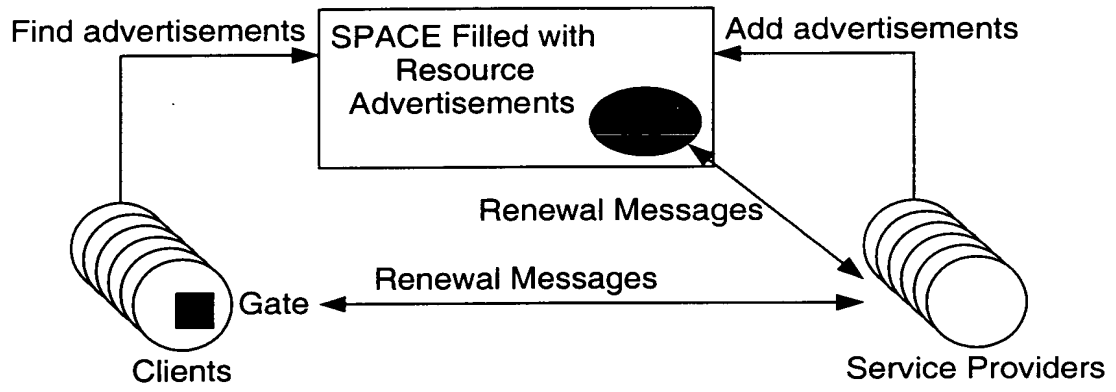


Fig. 25

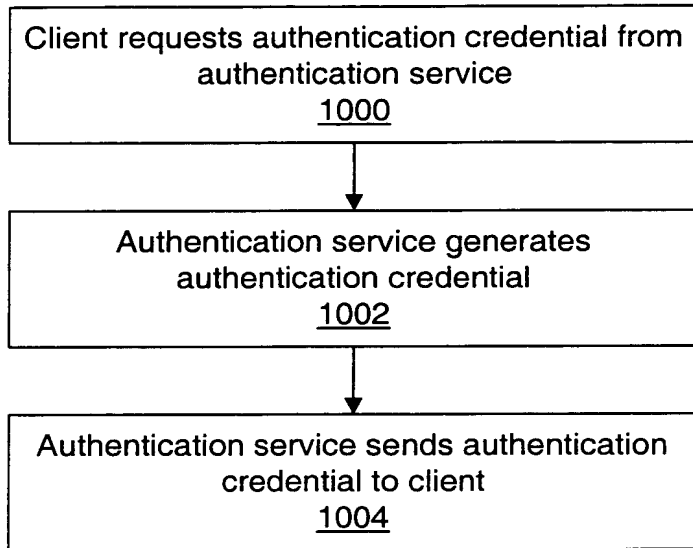


FIG. 26a

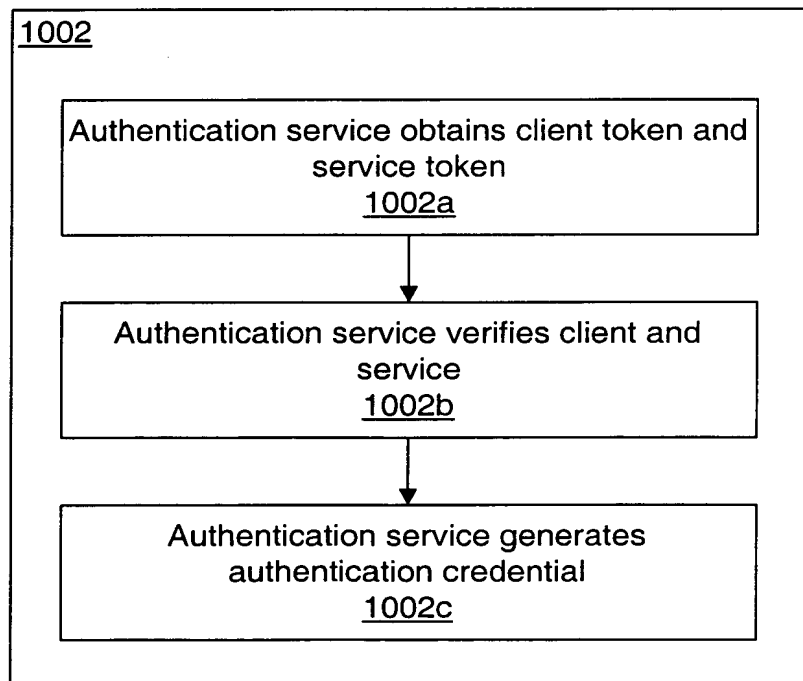


FIG. 26b

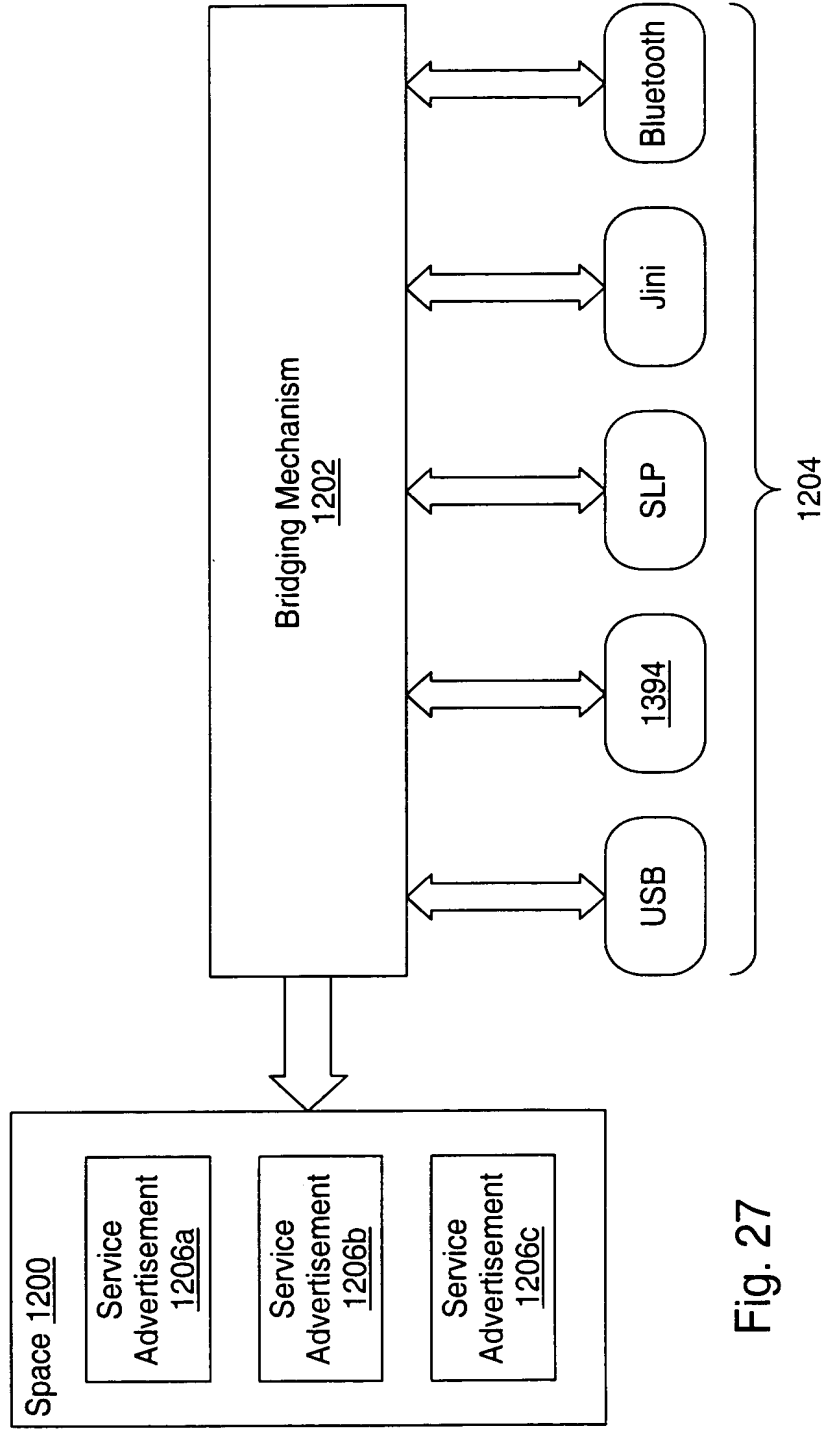


Fig. 27

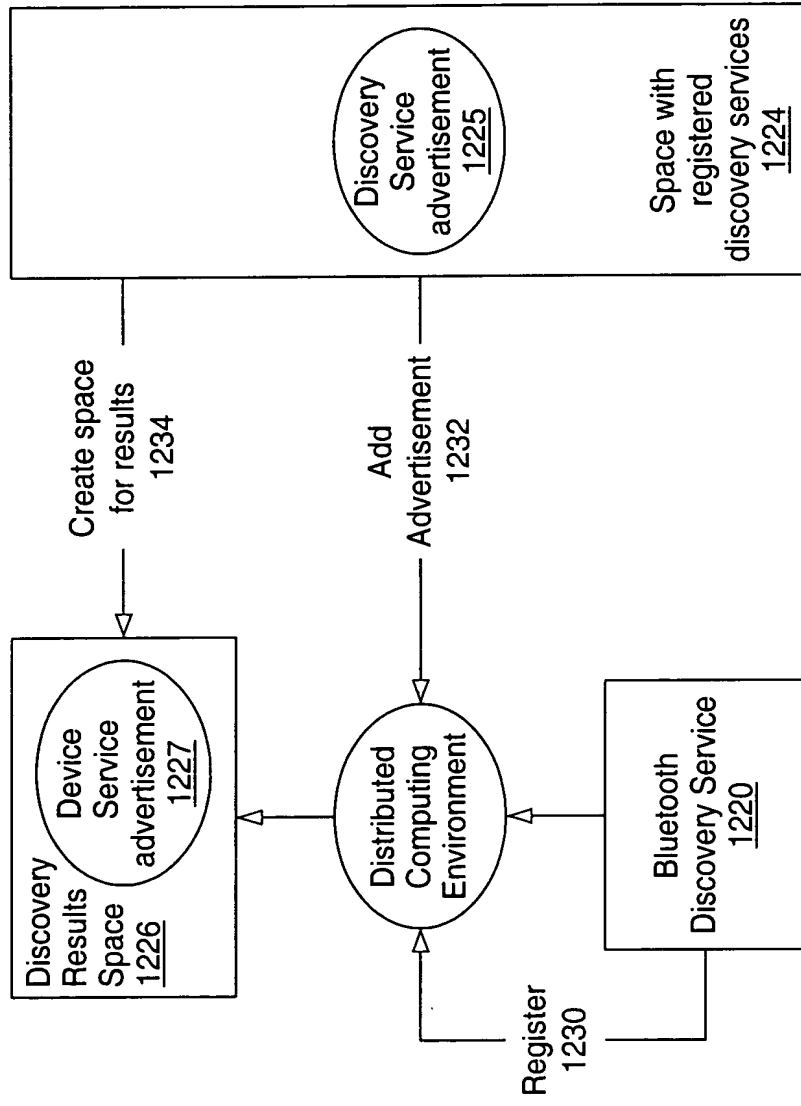


Fig. 28

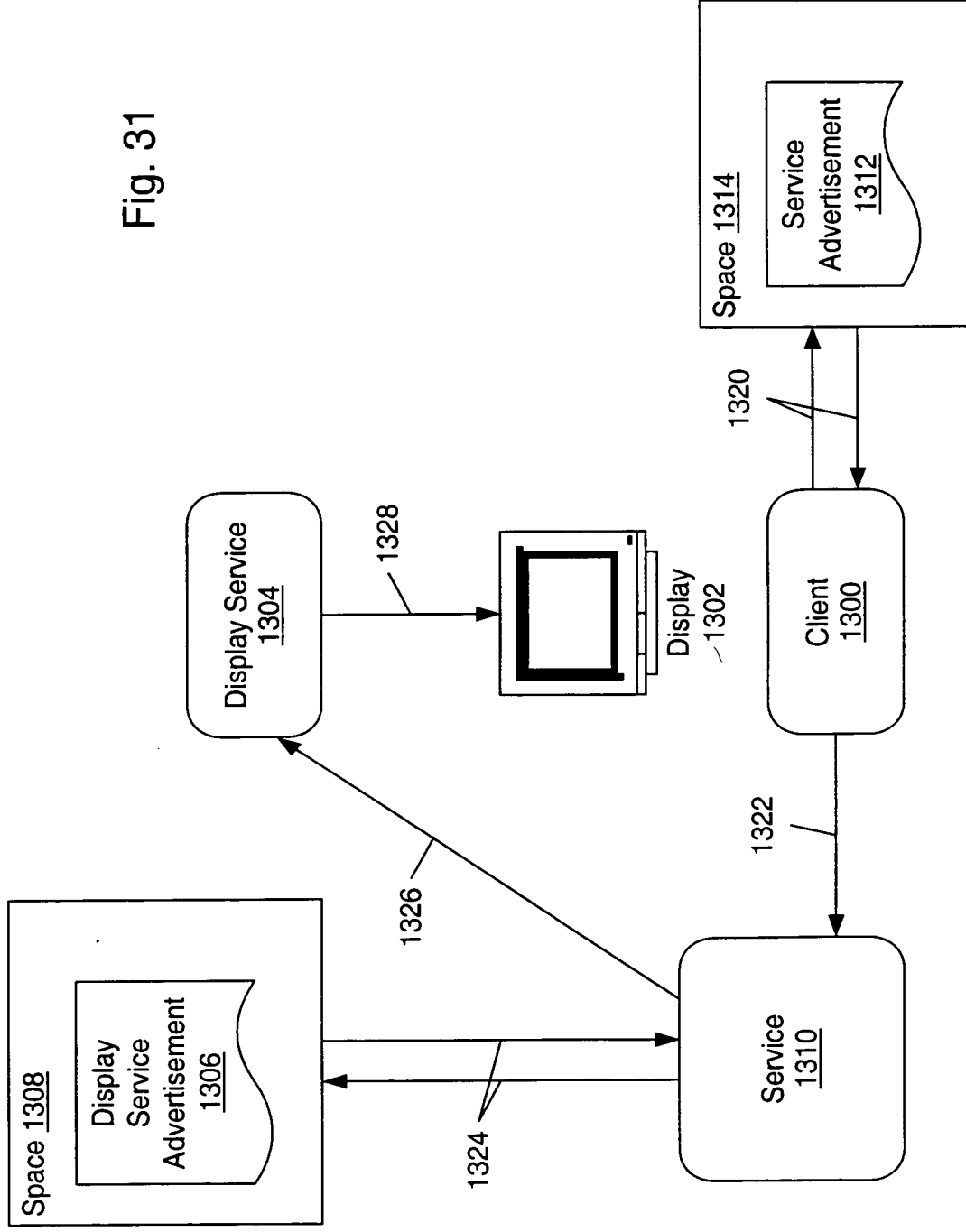


Fig. 31

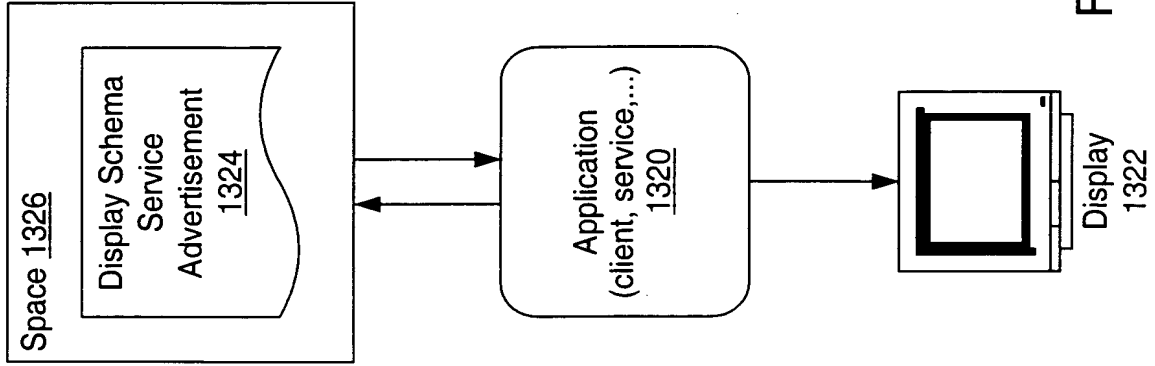


Fig. 32A

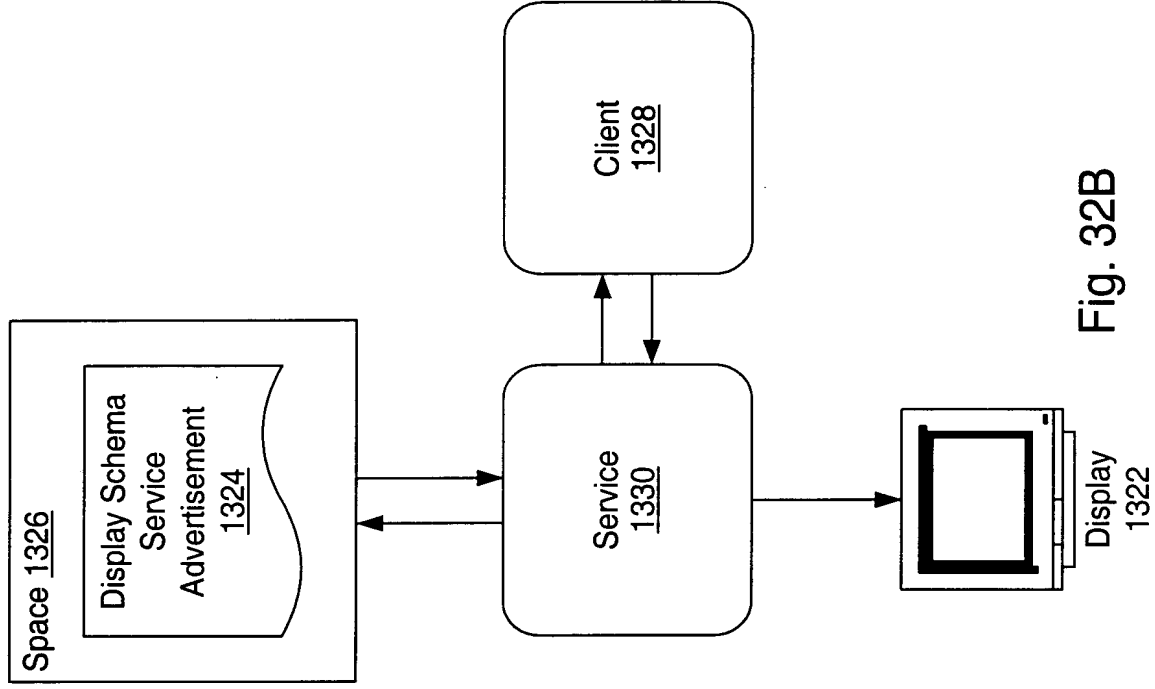


Fig. 32B

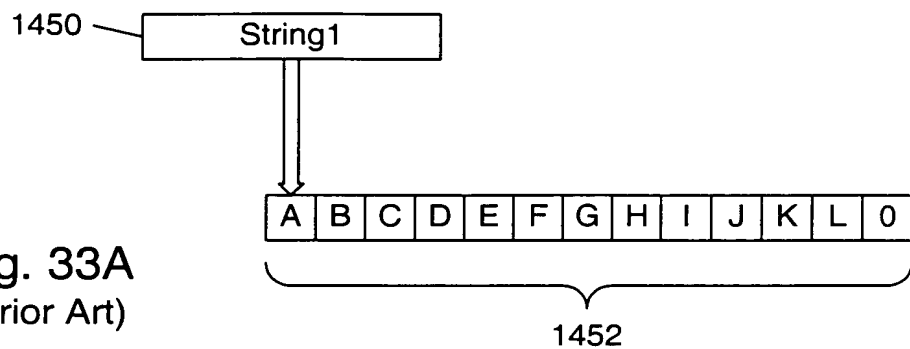
[illegible]

Fig. 33A
(Prior Art)

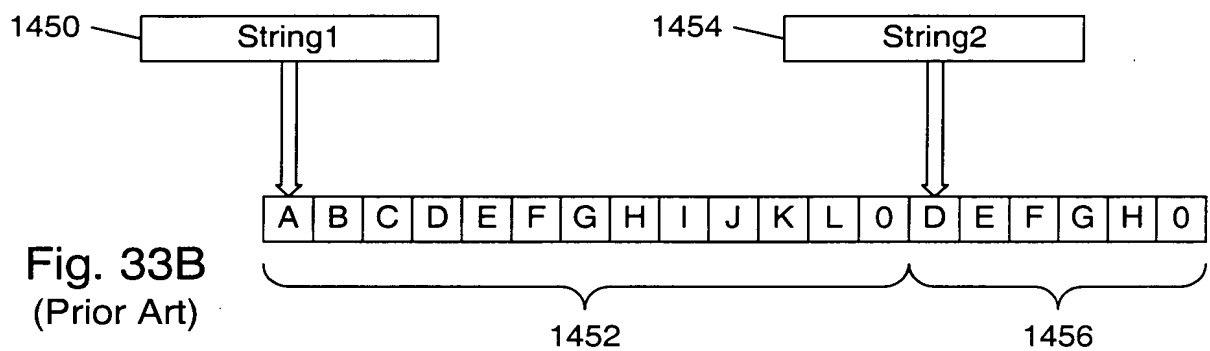


Fig. 33B
(Prior Art)

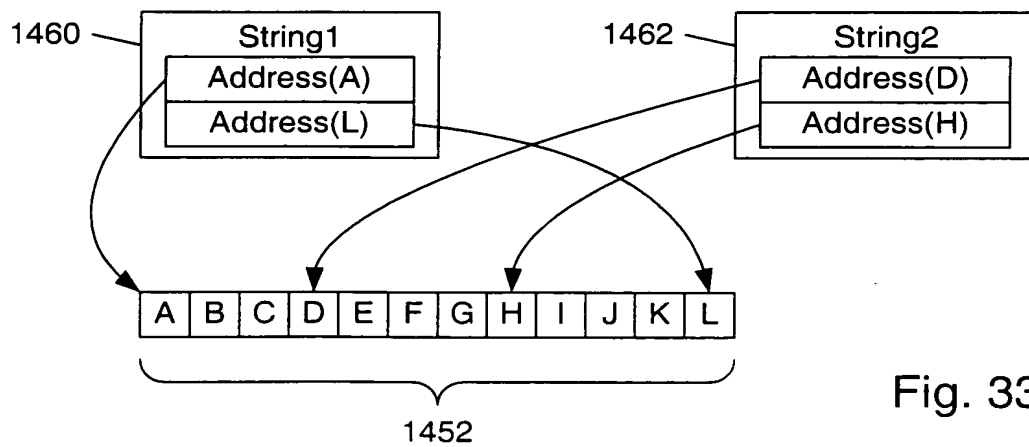


Fig. 33C

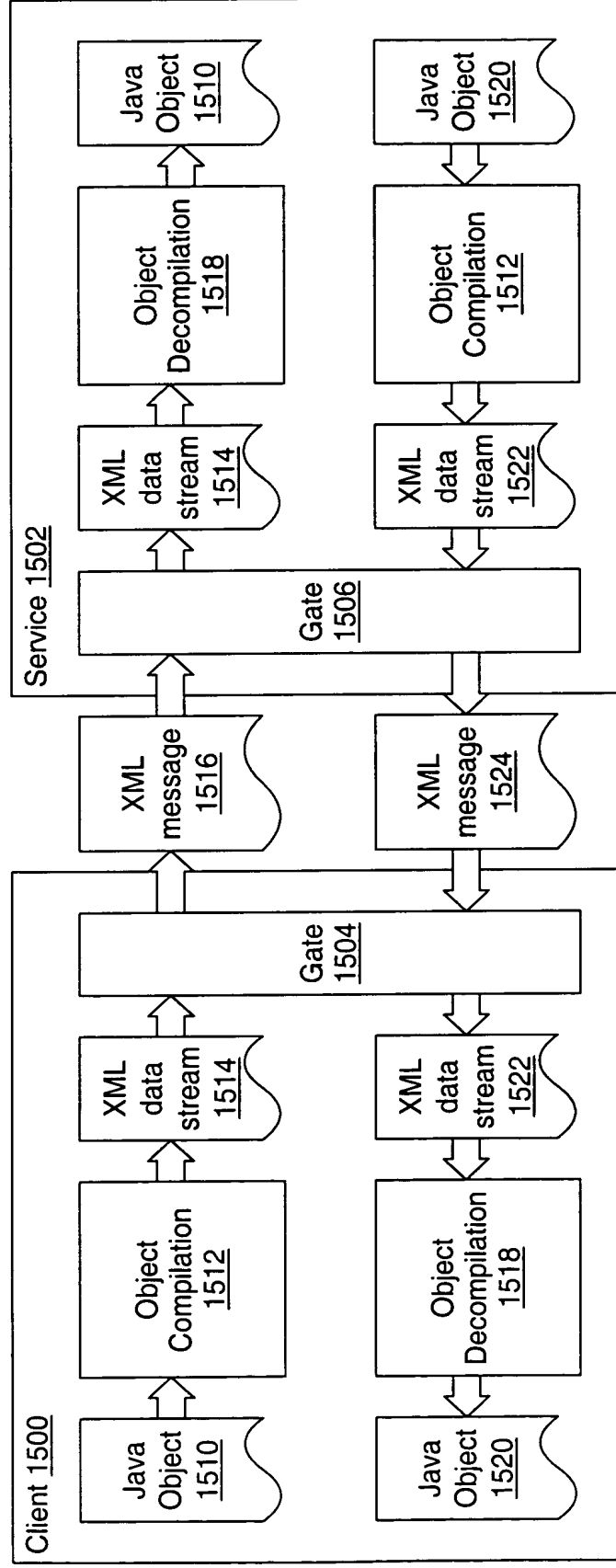


Fig. 34

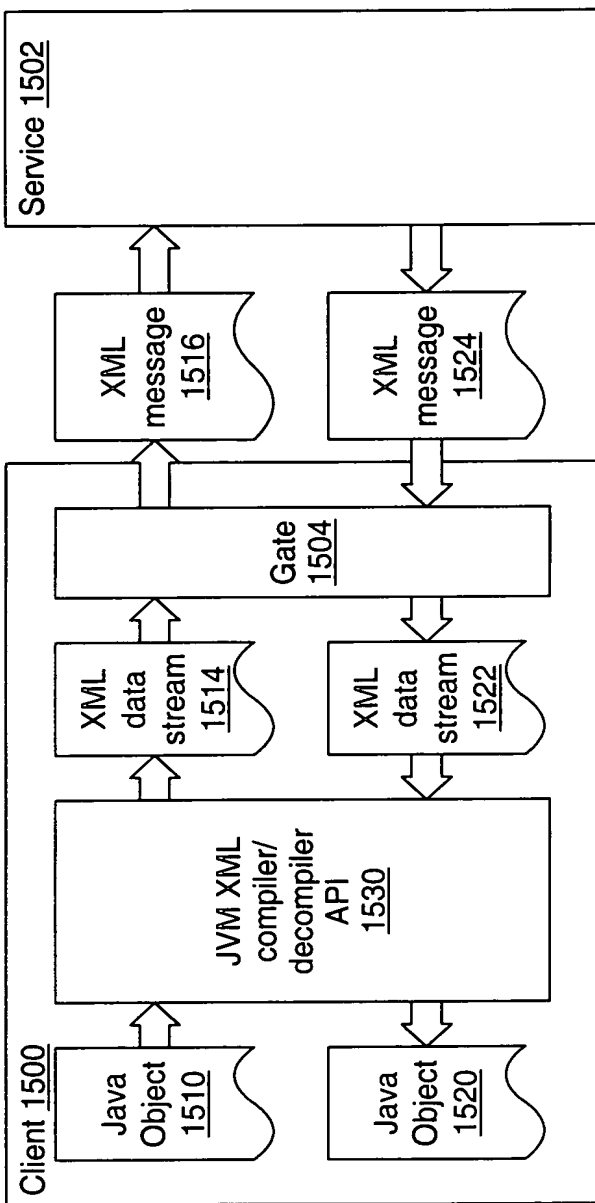


Fig. 35a

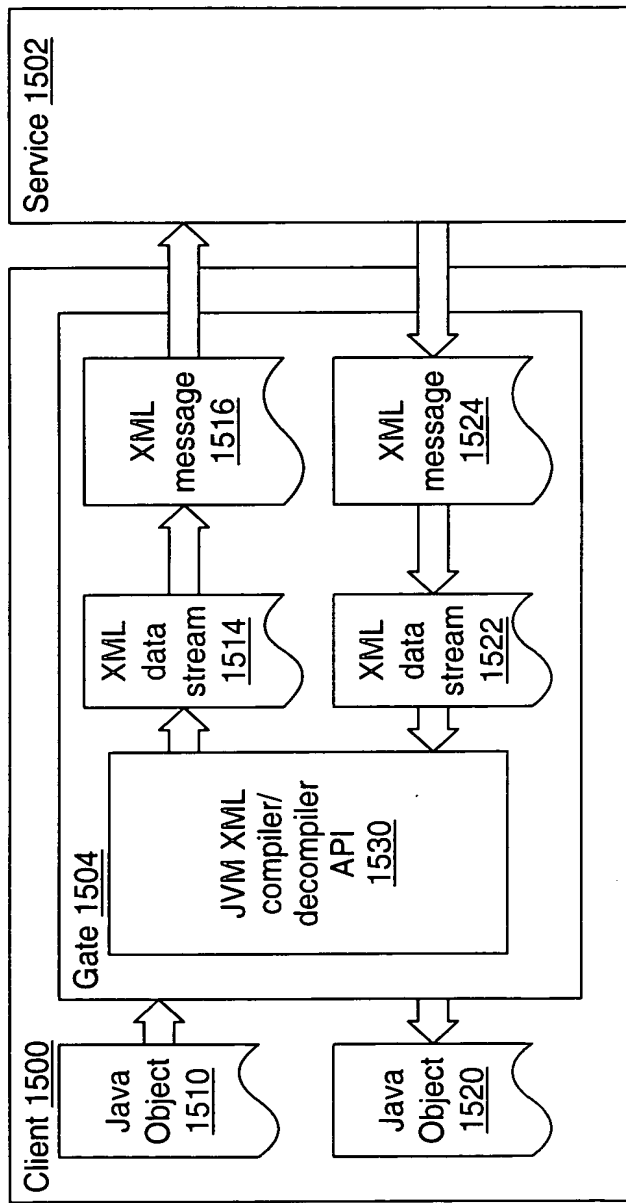


Fig. 35b

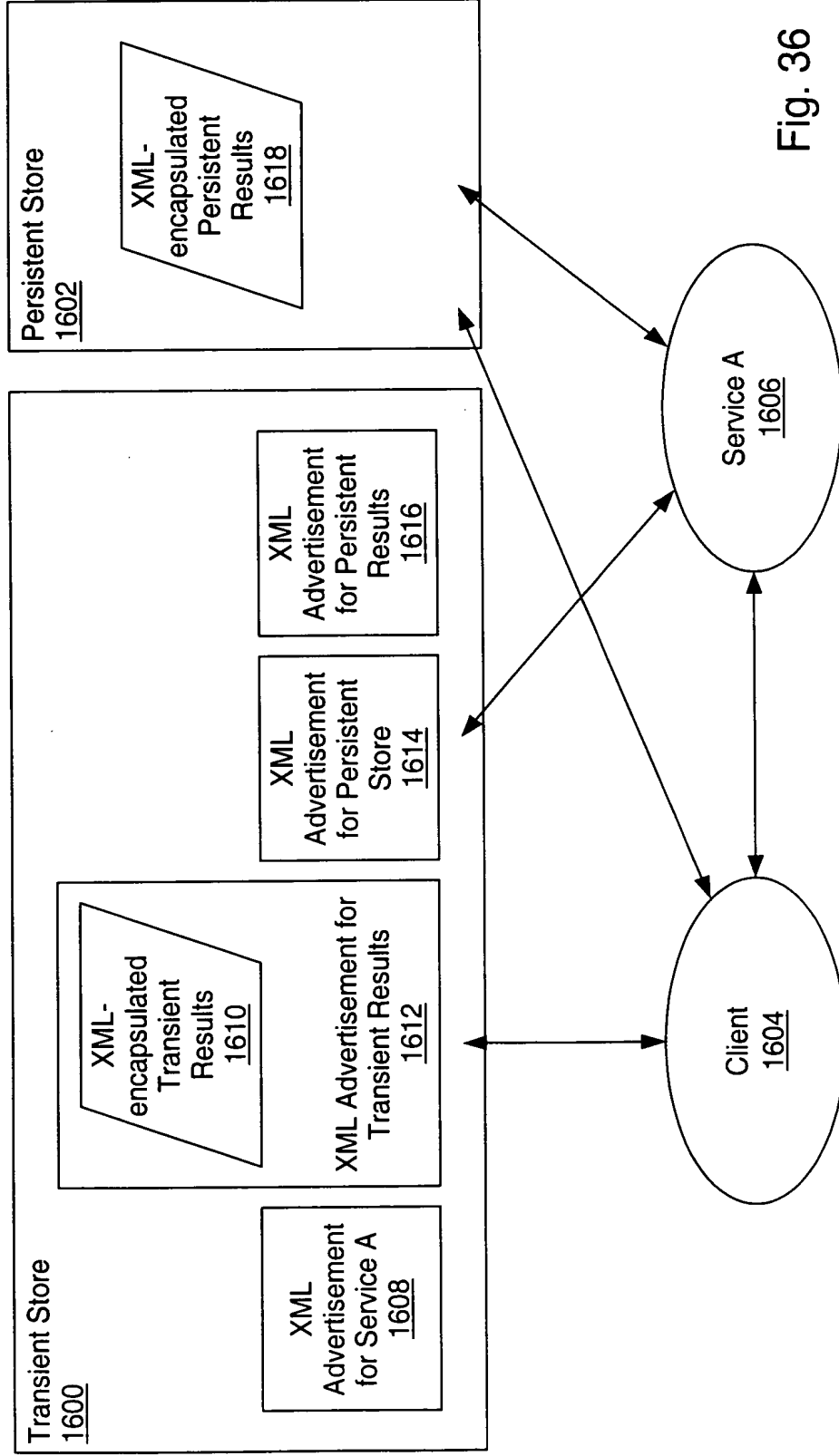


Fig. 36

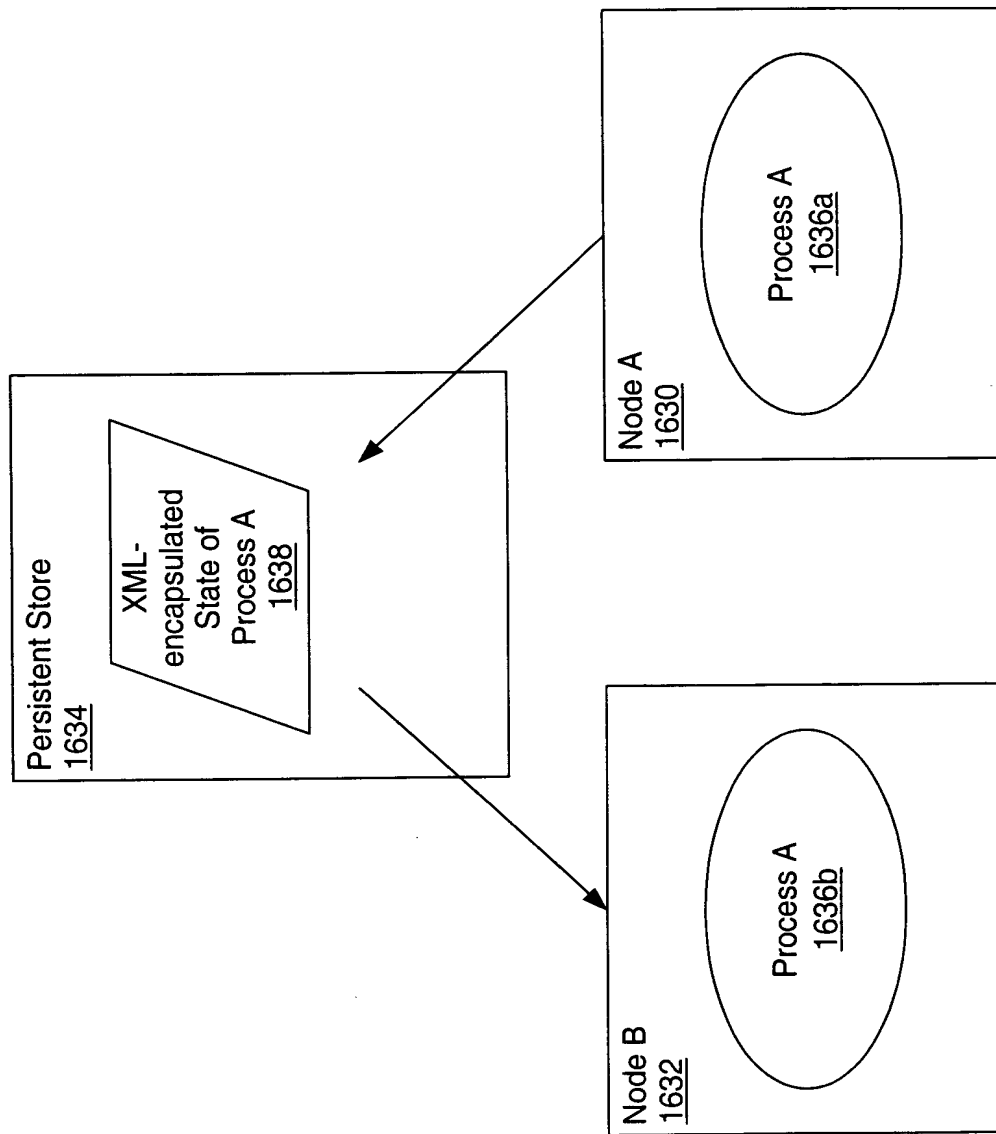


Fig. 37

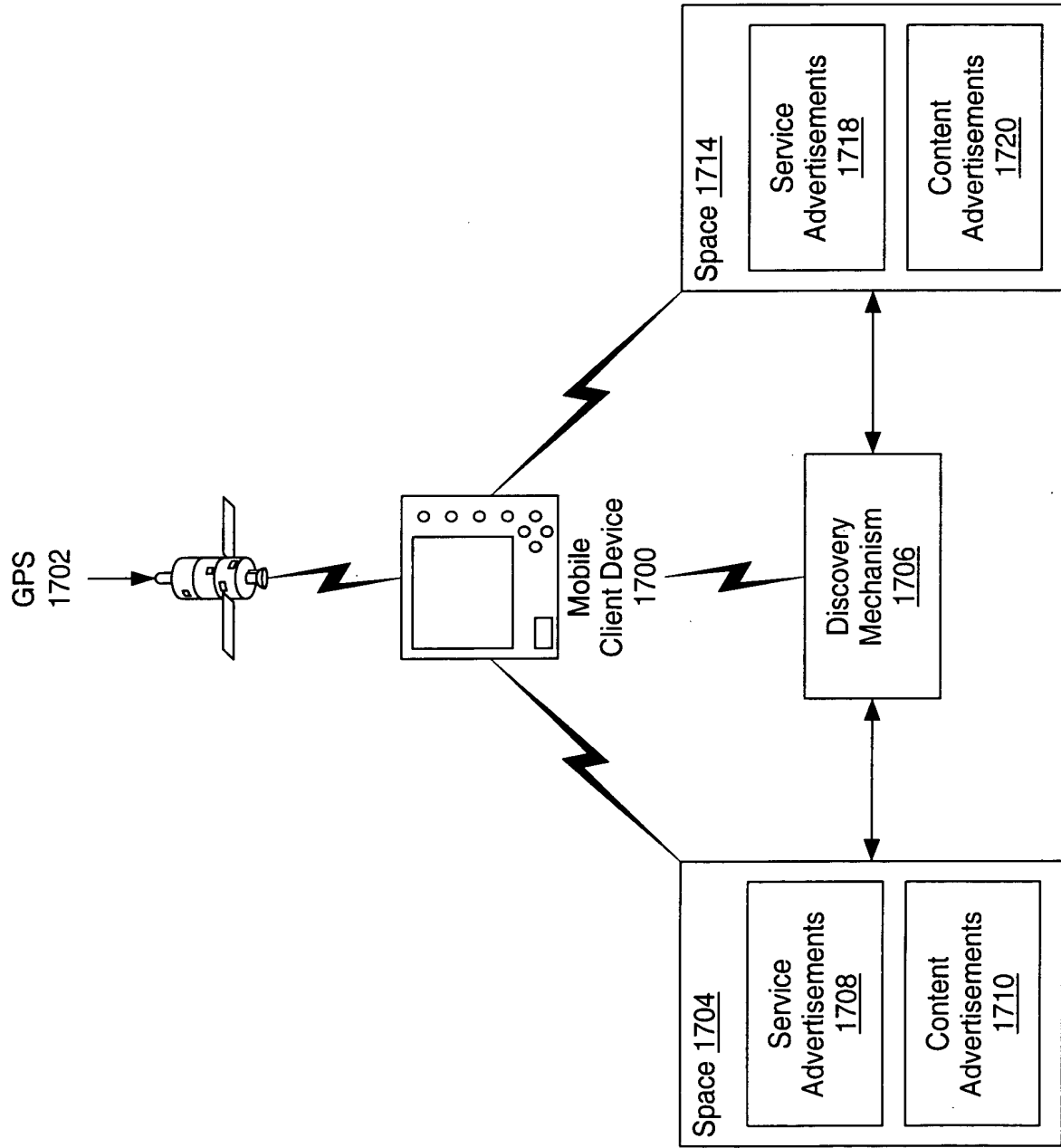


Fig. 38

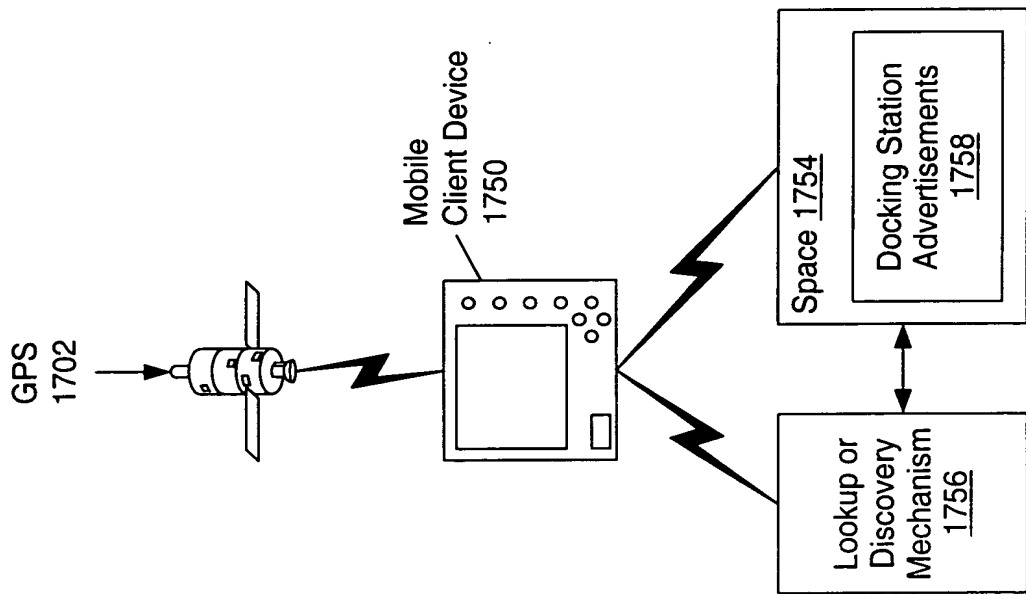


Fig. 39a

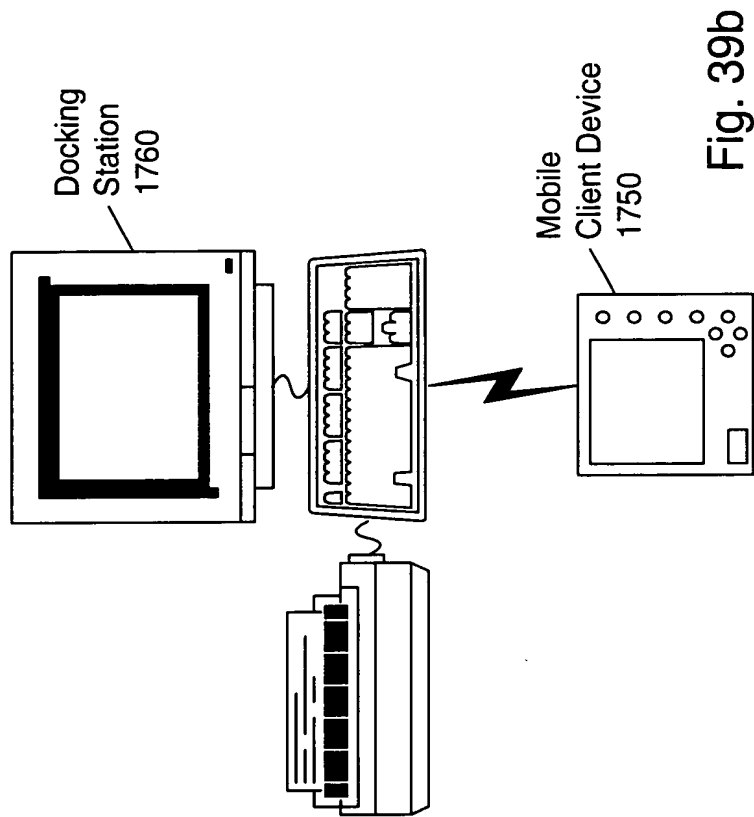


Fig. 39b

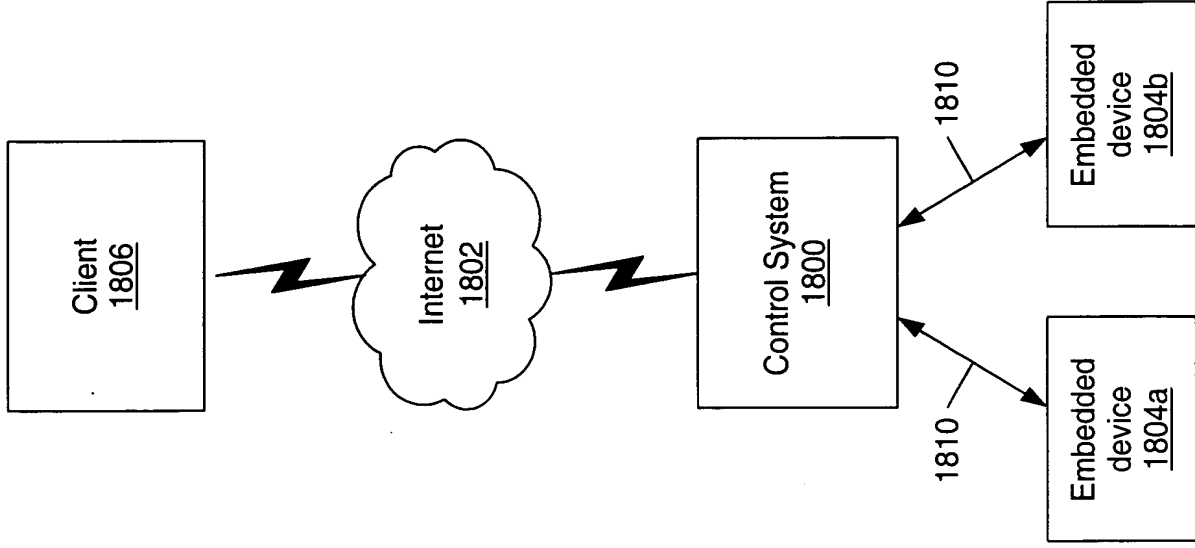


Fig. 40a

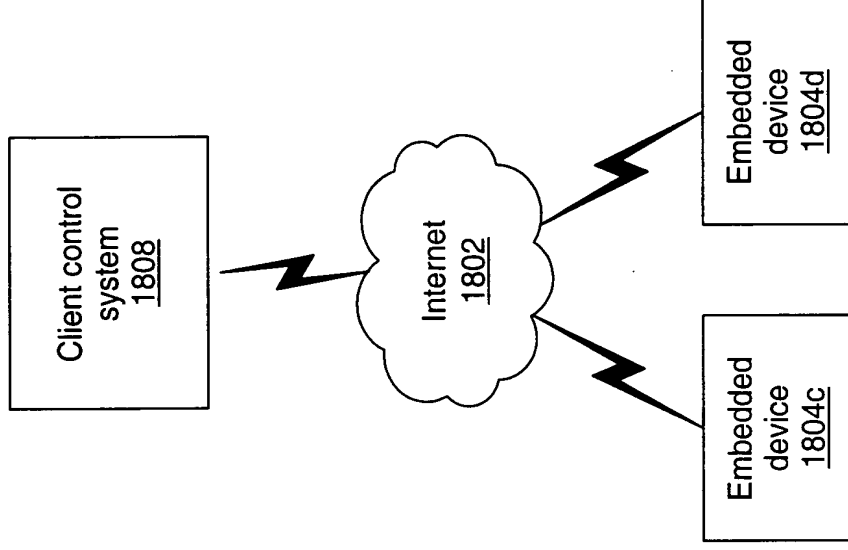


Fig. 40b

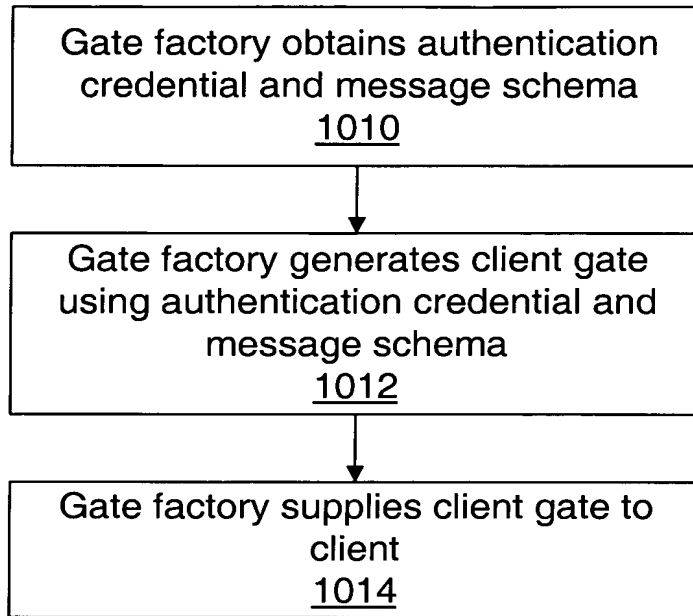


FIG. 41

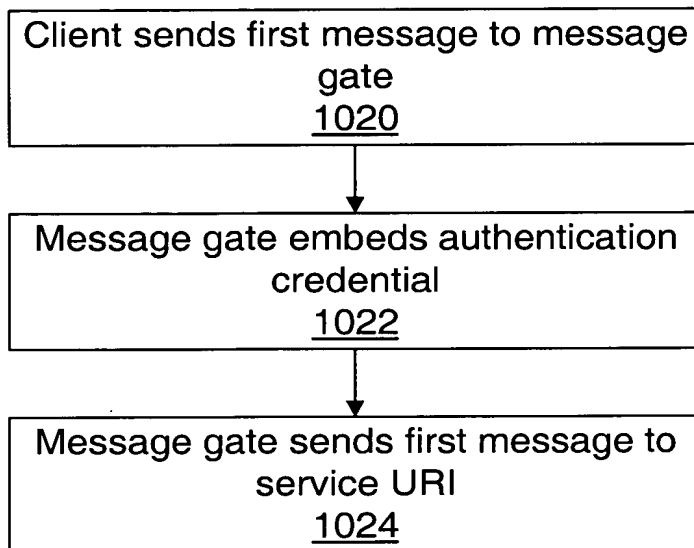


FIG. 42a

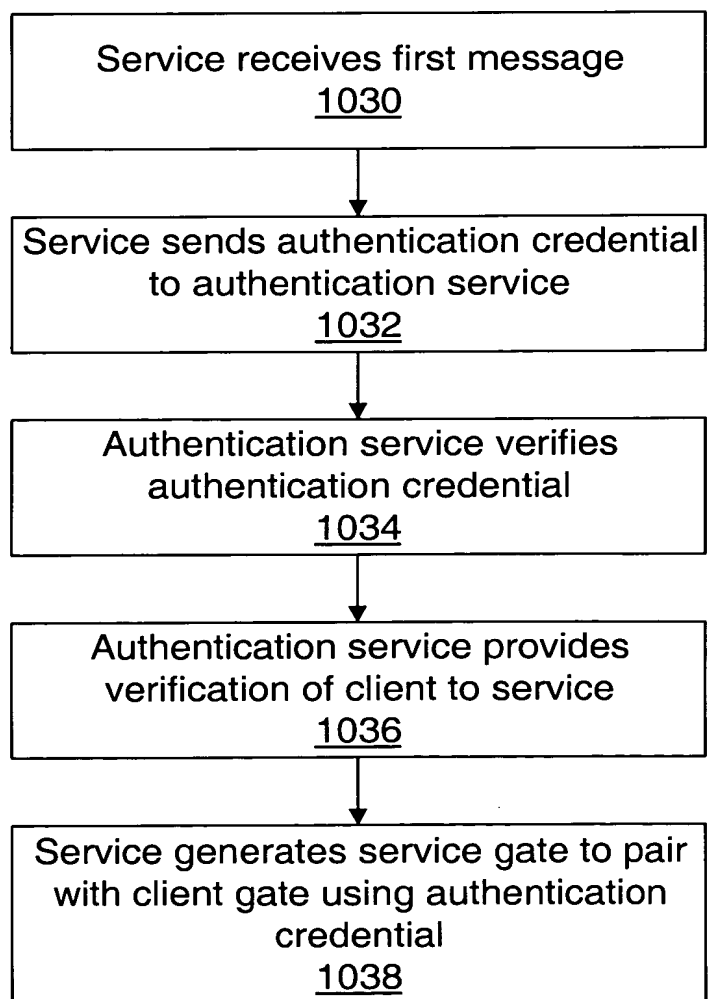


FIG. 42b

RECEIVED

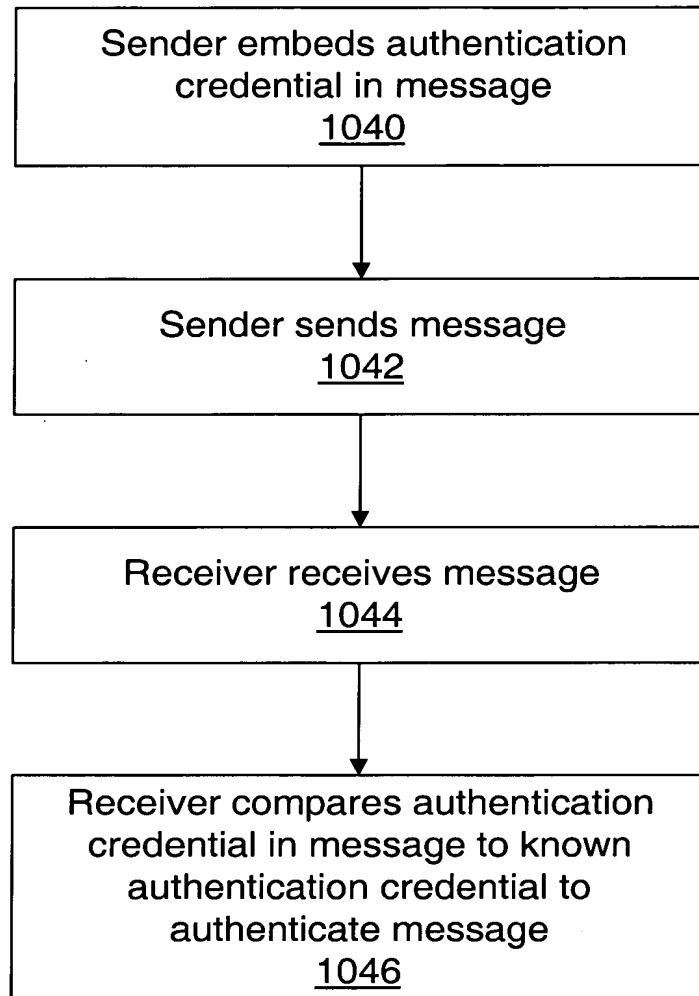


FIG. 42c

Sender embeds credential in message
1050

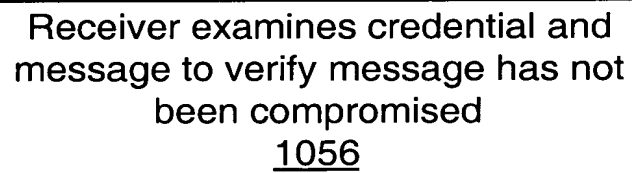
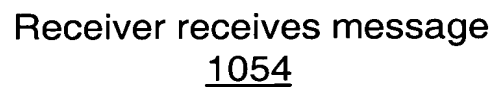
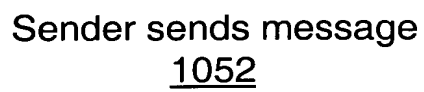


FIG. 43

Client requests lease on resource
provided by service
2000

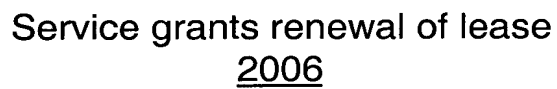
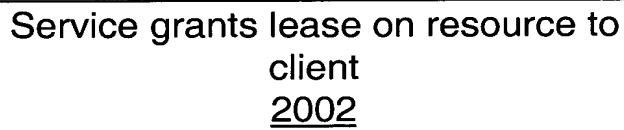


FIG. 44

```

graph TD
    2100[Client generates message including representation of method call  
2100] --> 2102[Client sends message to service  
2102]
    2102 --> 2104[Function performed on service based on representation of method call in the message  
2104]
    2104 --> 2106[Function generates results  
2106]
    2106 --> 2108[Results provided to client  
2108]

```

Figure 45a

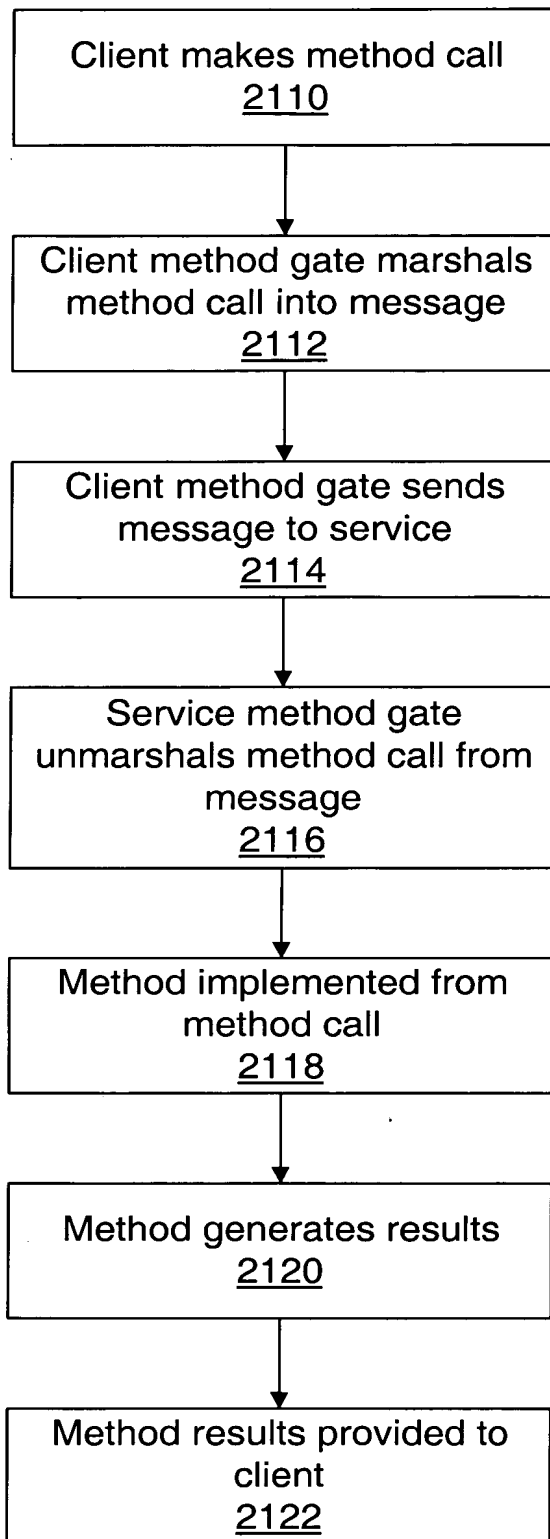


Figure 45b

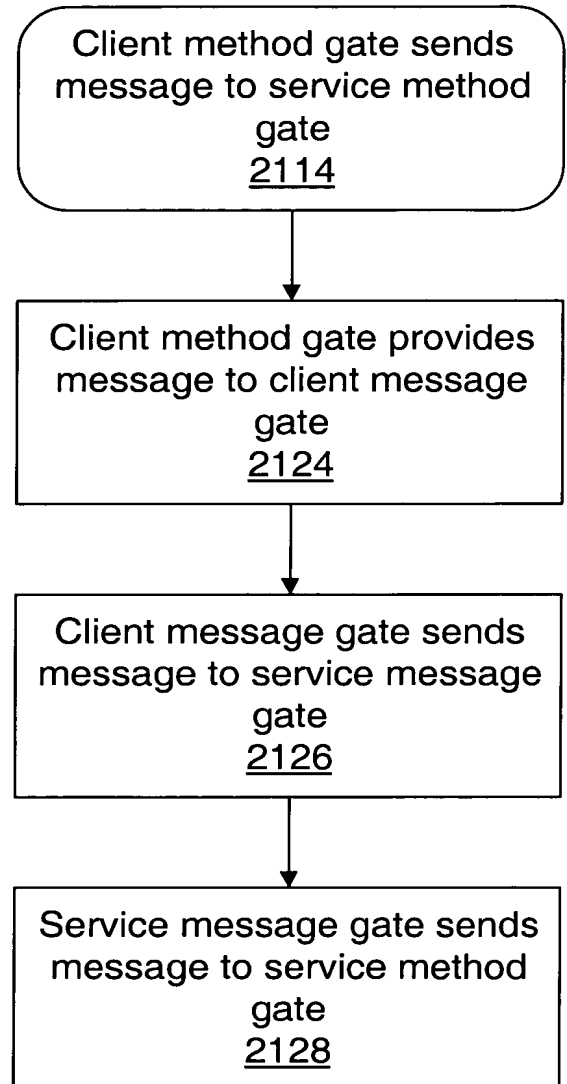


Figure 45c

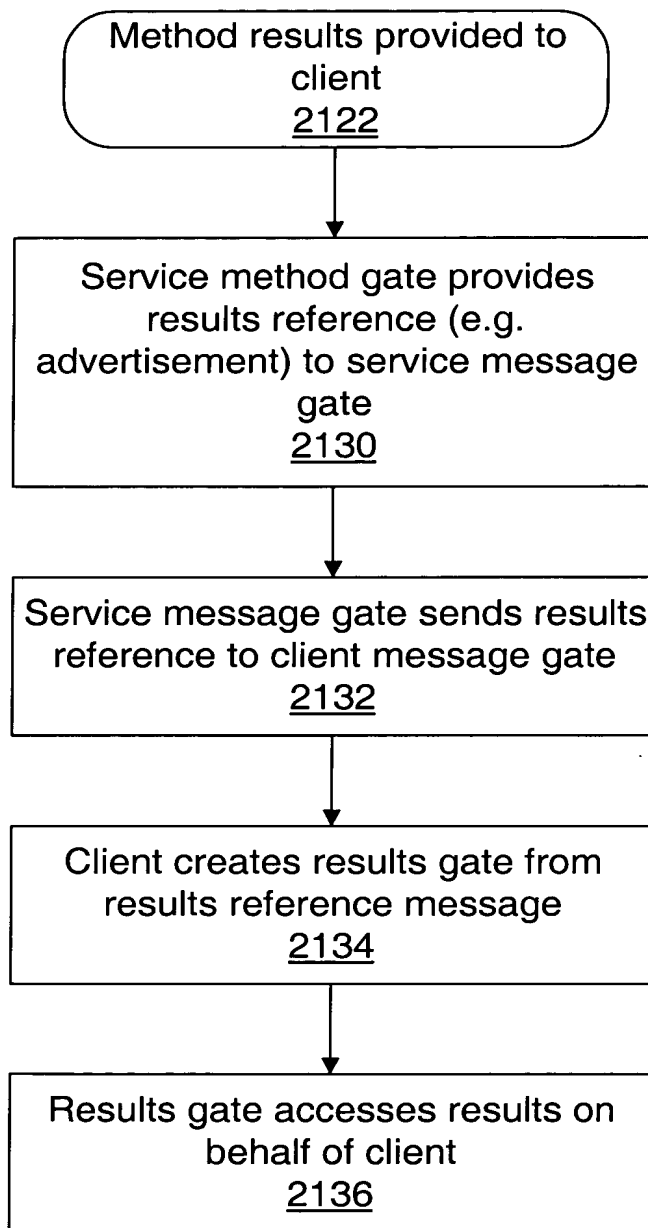


Figure 45d

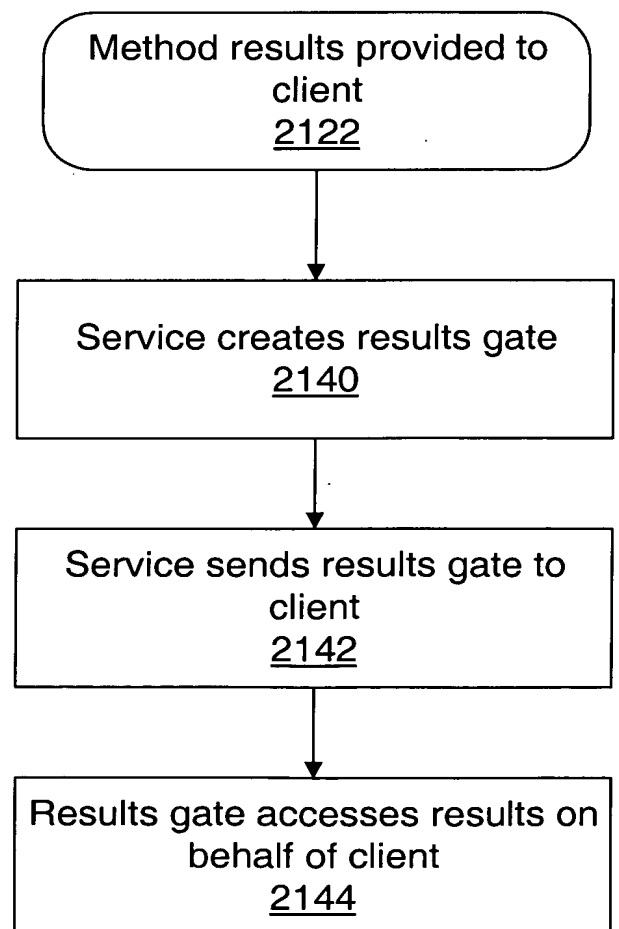


Figure 45e

```

graph TD
    2200[Client requests object from service  
2200] --> 2202[Service provides object to  
compilation process  
2202]
    2202 --> 2204[Compilation process generates data  
representation language  
representation of object  
2204]
    2204 --> 2206[Service sends data representation  
language representation of object to  
client in a message  
2206]
    2206 --> 2208[Client provides data representation  
language representation of object to  
decompilation process  
2208]
    2208 --> 2210[Decompilation process generates  
copy of object from data  
representation language  
representation of object  
2210]

```

Figure 46

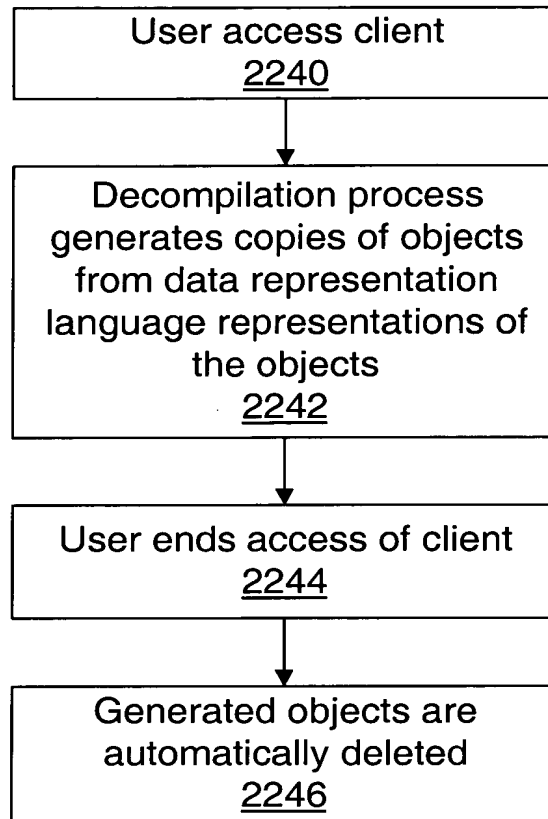


Figure 47

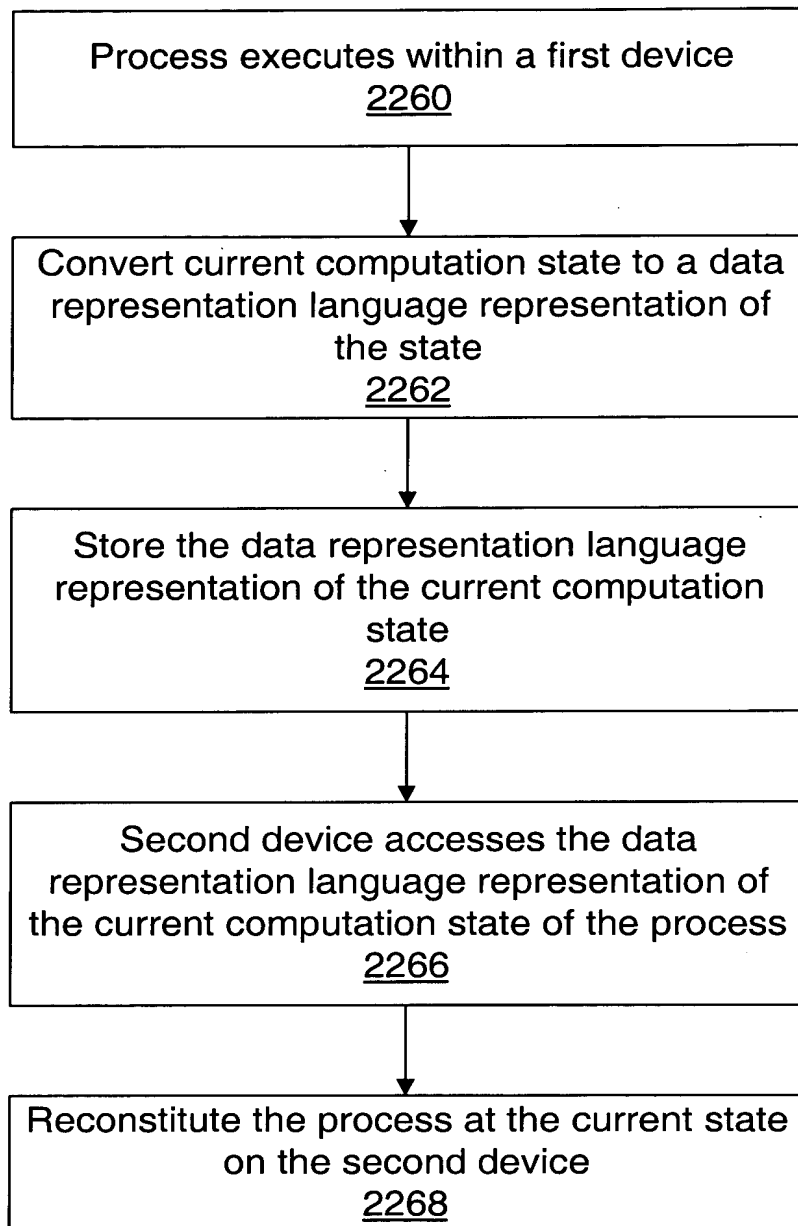


Figure 48

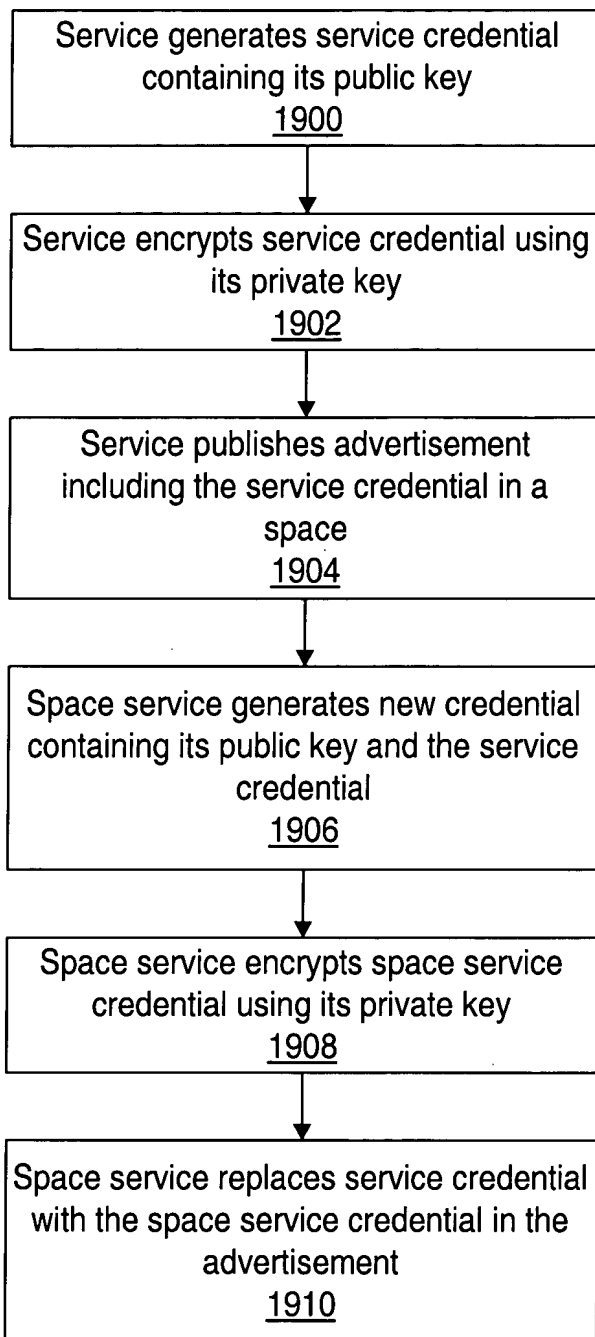


Fig. 49a

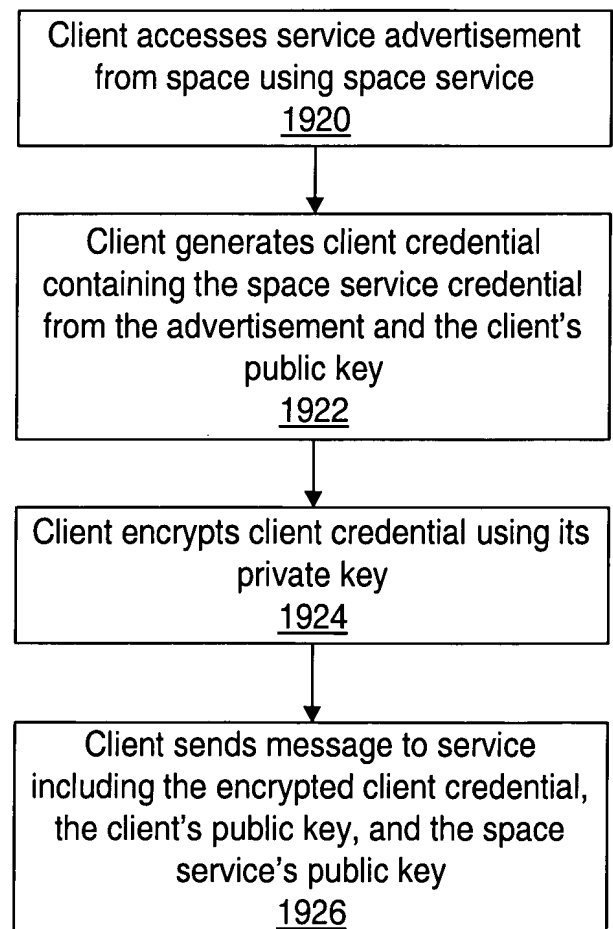


Fig. 49b

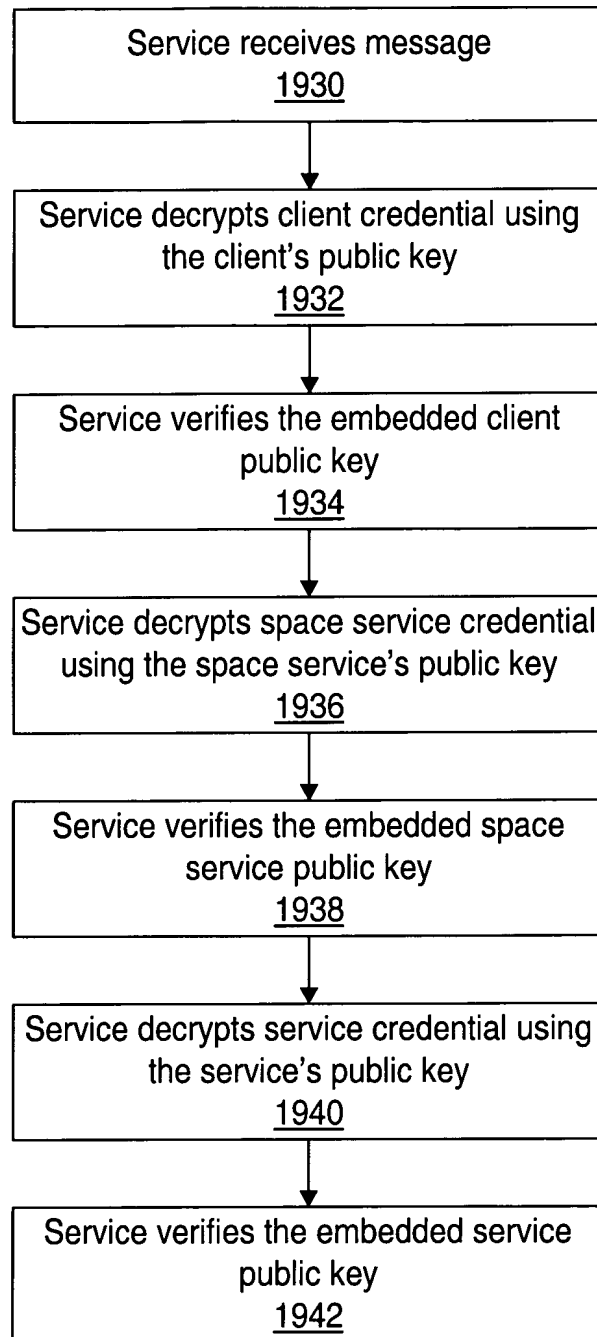
[illegible]

Fig. 49c

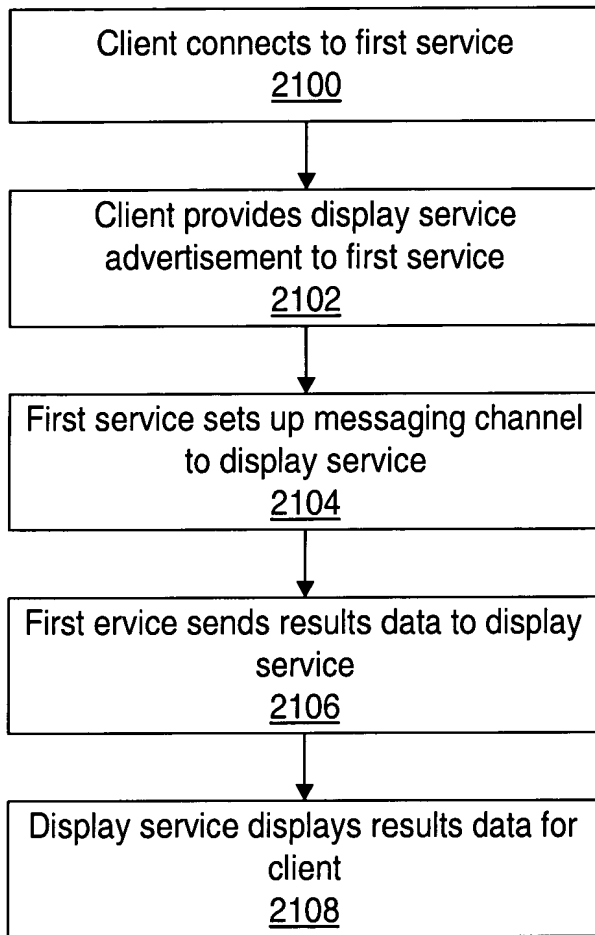


Fig. 50

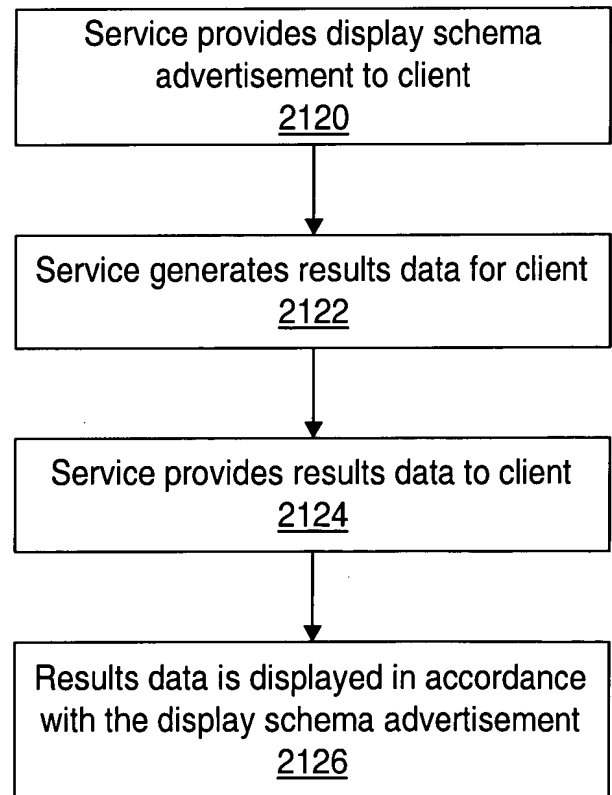


Fig. 51